Controller programmabili Twido

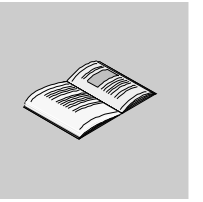
Guida di riferimento del software

TWD USE 10AE ita Version 3.2

a brand of Schneider Electric

Telemecanique

Indice



Informazioni di sicurezza 11

Informazioni su... 15

Parte I Descrizione del software Twido 17

In breve 17

Capitolo 1 Introduzione al software Twido 19

In breve 19
Introduzione a TwidoSoft. 20
Introduzione ai linguaggi Twido. 21

Capitolo 2 Oggetti del linguaggio Twido 25

In breve 25
Convalida degli oggetti 26
Oggetti bit 27
Oggetti parola 29
Oggetti a virgola mobile e parola doppia. 32
Indirizzamento di oggetti bit. 36
Indirizzamento di oggetti parola 37
Indirizzamento di oggetti a virgola mobile 38
Indirizzamento di oggetti parole doppie. 39
Indirizzamento degli I/O. 40
Indirizzamento di rete 43
Oggetti blocco funzione. 44
Oggetti strutturati. 46
Oggetti indicizzati 50
Oggetti di simbolizzazione 52

Capitolo 3 Memoria utente 53

In breve 53
Struttura della memoria utente 54
Backup e ripristino senza cartuccia di backup o memoria estesa. 56
Backup e ripristino con una cartuccia di backup da 32 K 58
Uso della cartuccia di memoria estesa da 64K. 61

Capitolo 4 Modalità operative del controller 65
In breve 65
Scansione ciclica 66
Scansione periodica. 68
Controllo del tempo di scansione. 71
Modalità operative 72
Gestione delle interruzioni e del ripristino dell'alimentazione 74
Gestione di un riavvio a caldo 76
Gestione di un avvio a freddo 78
Inizializzazione degli oggetti 80

Capitolo 5 Gestione dei task di evento 81
In breve 81
Presentazione dei task di evento. 82
Descrizione delle diverse sorgenti di evento 83
Gestione degli eventi 84

Parte II Funzioni speciali 85
In breve 85

Capitolo 6 Comunicazioni 87
In breve 87
Presentazione dei diversi tipi di comunicazione 88
Comunicazioni tra TwidoSoft e il controller 90
Comunicazioni tra TwidoSoft e un modem 96
Comunicazioni in collegamento remoto. 108
Comunicazioni ASCII. 120
Comunicazioni Modbus 132
Richieste standard Modbus 150
Classe di implementazione Transparent Ready (Twido Serial A05, Ethernet A15). . 155
Panoramica sulla comunicazione Ethernet TCP/IP 156
Guida alla configurazione TCP/IP rapida per la comunicazione Ethernet tra PC e controller . . 158
Connessione del controller alla rete. 164
Indirizzamento IP 165
Assegnazione di indirizzi IP 166
Configurazione TCP/IP 170
Scheda Configura indirizzo IP 172
Scheda IP contrassegnato. 175
Scheda Timeout. 177
Scheda Dispositivi remoti. 179
Visualizzazione della configurazione Ethernet. 181
Gestione delle connessioni Ethernet 182
Indicatori LED Ethernet 184
Messaggeria TCP Modbus 185

Capitolo 7 Incorporate, funzioni analogiche 189
In breve 189
Potenziometro analogico. 190
Canale analogico. 192

Capitolo 8 Gestione dei moduli analogici 193
In breve 193
Panoramica sui moduli analogici. 194
Indirizzamento degli I/O analogici 195
Configurazione degli I/O analogici. 196
Informazioni di stato del modulo analogico 202
Esempio di utilizzo dei moduli analogici 203

Capitolo 9 Implementazione del bus AS-Interface V2 205
In breve 205
Presentazione del bus AS-Interface V2 206
Descrizione funzionale generale 207
Principi di implementazione software 210
Descrizione della schermata di configurazione del bus AS-Interface 211
Configurazione del bus AS-Interface. 213
Descrizione della schermata di debug 219
Modifica dell'indirizzo di uno slave 222
Aggiornamento della configurazione del bus AS-Interface in modalità collegato. . . 224
Indirizzamento automatico di uno slave AS-Interface V2 229
Inserimento di un'apparecchiatura slave in una configurazione AS-Interface V2 esistente. . . .230
Sostituzione automatica di uno slave AS-Interface V2 in errore 231
Indirizzamento degli ingressi/uscite associati alle apparecchiature slave collegate sul bus AS-Interface V2. 232
Programmazione e diagnostica del bus AS-Interface V2 234
Modalità di funzionamento del modulo interfaccia bus AS-Interface V2 238

Capitolo 10 Installazione e configurazione del bus di campo CANopen . 239
In breve 239
10.1 Panoramica del bus di campo CANopen 241
In breve 241
Elementi di base CANopen 242
Informazioni su CANopen 243
CANOpen Boot-Up 246
Process Data Object (PDO) - Trasmissione 249
Accesso ai dati tramite scambi espliciti (SDO) 251
"Node Guarding e Life Guarding 252
Gestione del bus interno 254
10.2 Implementazione del bus CANopen 255
Panoramica 255
Panoramica 256

Configurazione hardware 257
Metodologia di configurazione 258
Dichiarazione del master CANopen 260
Dichiarazione sulla rete di slave CANopen 261
Mappatura oggetti CANopen 265
Collegamento oggetti CANopen 269
Rappresentazione simbolica degli oggetti CANopen 271
Indirizzamento dei PDO del master CANopen 273
Programmazione e diagnostica del bus di campo CANopen 274

Capitolo 11 Configurazione del gateway Ethernet TwidoPort 281
In breve 281
11.1 Configurazione normale e connessione di TwidoPort 283
In breve 283
Configurazione normale con TwidoSoft 284
Configurazione BootP 289
11.2 Configurazione Telnet di TwidoPort 290
In breve 290
Introduzione alla configurazione Telnet 291
Menu principale di Telnet 292
Impostazioni IP/Ethernet 293
Configurazione dei parametri seriali 294
Configurazione del gateway 295
Configurazione della sicurezza 296
Statistiche Ethernet 297
Statistiche seriali 298
Salvataggio della configurazione 299
Ripristino delle impostazioni predefinite 300
Aggiornamento del firmware TwidoPort 301
Se si dimentica la password e/o la configurazione IP 303
11.3 Funzioni di comunicazione 304
In breve 304
Funzioni Ethernet 305
Protocollo di comunicazione Modbus/TCP 306
Codici funzione Modbus supportati localmente 307

Capitolo 12 Funzionamento del visualizzatore 309
In breve 309
Pannello visualizzatore 310
Informazioni di identificazione e di stato del controller 313
Variabili e oggetti di sistema 315
Impostazioni delle porte seriali 322
Orologio in tempo reale 323
Fattore di correzione dell'orologio-datario 324

Parte III Descrizione dei linguaggi Twido 325
In breve 325

Capitolo 13 Ladder, linguaggio 327
In breve 327
Introduzione ai linguaggi Ladder 328
Principi di programmazione dei linguaggi Ladder 330
Blocchi linguaggio Ladder 332
Elementi grafici linguaggio Ladder 335
Istruzioni Ladder speciali OPEN e SHORT 338
Note per la programmazione 339
Reversibilità Ladder/List 343
Indicazioni per la reversibilità Ladder/List 344
Documentazione del programma 346

Capitolo 14 Lista d'istruzioni, linguaggio 349
In breve 349
Panoramica dei programmi List 350
Funzionamento delle istruzioni List 352
Istruzioni linguaggio List 353
Utilizzo delle parentesi 356
Istruzioni Stack (MPS, MRD, MPP) 358

Capitolo 15 Grafcet 361
In breve 361
Descrizione delle istruzioni Grafcet 362
Descrizione della struttura di programma Grafcet 366
Azioni associate ai passi Grafcet 369

Parte IV Descrizione di istruzioni e funzioni 371
In breve 371

Capitolo 16 Istruzioni di base 373
In breve 373
16.1 Elaborazione booleana 375
In breve 375
Istruzioni booleane 376
Informazioni sul formato per la descrizione delle istruzioni booleane 378
Istruzioni Load (LD, LDN, LDR, LDF) 380
Istruzioni d'assegnazione (ST, STN, R, S) 382
Istruzioni AND logico (AND, ANDN, ANDR, ANDF) 384
Istruzioni OR logico (OR, ORN, ORR, ORF) 386
OR esclusivo, istruzioni (XOR, XORN, XORR, XORF) 388
Istruzione NOT (N) 390
16.2 Blocchi funzione di base 392

In breve 392
Blocchi funzione di base 393
Principi di programmazione dei blocchi funzione standard 395
Blocco funzione timer (%TMi) 397
Timer di tipo TOF 399
Timer di tipo TON 400
Tipo di timer TP 401
Programmazione e configurazione timer 402
Blocco funzione contatore avanti/indietro (%Ci) 405
Programmazione e configurazione contatori 408
Blocco funzione registro bit di scorrimento (%SBRi) 410
Blocco funzione contatore passo passo (%SCi) 412
16.3 Elaborazione numerica 415
In breve 415
Introduzione alle istruzioni digitali 416
Istruzioni di assegnazione 417
Istruzioni di confronto 422
Istruzioni aritmetiche su interi 424
Istruzioni logiche 428
Istruzioni di scorrimento 429
Istruzioni di conversione 431
Istruzioni di conversione tra parole semplici e doppie 433
16.4 Istruzioni di programma 434
In breve 434
Istruzioni END 435
Istruzione NOP 437
Istruzioni di salto 438
Istruzioni di subroutine 439

Capitolo 17 Istruzioni avanzate 441
In breve 441
17.1 Blocchi di funzione avanzati 443
In breve 443
Oggetti bit e parola associati ai blocchi funzione avanzati 444
Principi di programmazione dei blocchi funzione avanzati 446
Blocco funzione registro LIFO/FIFO (%Ri) 448
Operazione LIFO 449
FIFO, funzionamento 450
Programmazione e configurazione dei registri 451
Blocco funzione Modulazione d'ampiezza d'impulso (%PWM) 454
Blocco funzione uscita generatore d'impulsi (%PLS) 457
Blocco funzione del Controller a tamburo (%DR) 460
Operazione %DRi del blocco funzione Controller a tamburo 461
Programmazione e configurazione dei programmatori a tamburo 463
Blocco funzione contatore veloce (%FC) 465
Blocco funzione contatore molto veloce (%VFC) 468

Trasmissione e ricezione di messaggi – Istruzione di scambio (EXCH) 479
Blocco funzione controllo di scambio (%MSGx) 480
17.2 Funzioni dell'orologio. 483
In breve 483
Funzioni dell'orologio. 484
Blocchi orologio-datario. 485
Timbro data/ora. 488
Impostazione della data e dell'ora. 490
17.3 Guida all'avvio rapido della funzione PID Twido 493
In breve 493
Scopo del documento 494
Fase 1 - Configurazione dei canali analogici utilizzati per la regolazione 496
Fase 2 - Prerequisiti della configurazione del PID 498
Fase 3 - Configurazione del PID 500
Fase 4 - Inizializzazione dell'implementazione della regolazione 507
Fase 5 - Implementazione della regolazione AT + PID. 511
Fase 6 - Debug delle regolazioni. 515
17.4 Funzione PID. 518
In breve 518
Panoramica 519
Principio del loop di regolazione 520
Metodologia di sviluppo di un'applicazione di regolazione 521
Compatibilità e prestazioni 522
Caratteristiche dettagliate della funzione PID 523
Accesso alla configurazione del PID. 526
Scheda Generale del PID 528
Scheda Ingresso del PID. 531
Scheda PID del PID 533
Scheda AT della funzione PID 535
Scheda Uscita del PID 540
Accesso al debug del PID. 543
Scheda Animazione del PID 545
Scheda Traccia del PID. 547
Stati PID e codici di errore. 549
Sintonizzazione del PID con sintonizzazione automatica (Auto-Tuning, AT) . . 553
Metodo di regolazione dei parametri PID 561
Ruolo e influenza dei parametri di un PID. 564
Appendice 1: Fondamenti della teoria PID 568
Appendice 2: Modello di prim'ordine con ritardo 570
17.5 Istruzioni su virgola mobile 572
In breve 572
Istruzioni aritmetiche in virgola mobile 573
Istruzioni trigonometriche 576
Istruzioni di conversione 578
Istruzioni di conversione interi <-> Virgola mobile. 580

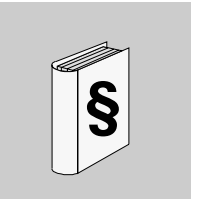
17.6 Istruzioni sulle tabelle di oggetti 583
In breve 583
Funzione di somma sulle tabelle 584
Funzione di confronto delle tabelle 586
Funzioni di ricerca sulle tabelle 588
Funzioni di ricerca del valore massimo e minimo sulle tabelle 590
Numero di occorrenze di un valore in una tabella 591
Funzione di scorrimento circolare su una tabella 592
Funzione di ordinamento sulle tabelle 594
Funzione di interpolazione su una tabella a virgola mobile 596
Funzione di calcolo della media dei valori di una tabella a virgola mobile 601

Capitolo 18 Bit di sistema e parole di sistema 603
In breve 603
Bit di sistema (%S). 604
Parole di sistema (%SW). 614

Glossario 627

Indice analitico 641

Informazioni di sicurezza



Informazioni importanti

AVVISO

Leggere attentamente queste istruzioni e osservare l'apparecchiatura per acquisire familiarità prima di iniziare le operazioni di installazione, uso o manutenzione. I seguenti messaggi speciali possono apparire in diverse parti della documentazione oppure essere indicati sull'apparecchiatura per segnalare rischi o per richiamare l'attenzione su informazioni che chiariscono o semplificano una procedura.



L'aggiunta di questo simbolo all'etichetta di un segnale di pericolo o di avvertenza indica che esiste un rischio da shock elettrico per l'incolumità personale se non vengono rispettate le istruzioni.



Questo è il simbolo di un allarme di sicurezza che viene utilizzato per segnalare rischi di potenziali lesioni personali. Rispettare tutti gli avvisi di sicurezza provvisti di questo simbolo al fine di evitare lesioni o rischi all'incolumità personale.

⚠ PERICOLO

PERICOLO indica una condizione immediata di pericolo, la quale, se non evitata, **può creare** gravi rischi all'incolumità personale o danni alle apparecchiature.

⚠ AVVERTENZA

AVVERTENZA indica una situazione potenziale di pericolo, la quale, se non evitata, **può causare** gravi rischi all'incolumità personale o danni alle apparecchiature.

⚠ ATTENZIONE

ATTENZIONE indica una situazione potenziale di pericolo, la quale, se non evitata, **può causare** rischi all'incolumità personale o danni alle apparecchiature.

NOTA

La manutenzione alle apparecchiature elettriche deve essere eseguita solo da personale qualificato. Schneider Electric non assume alcuna responsabilità per qualunque conseguenza derivante dall'uso di questo materiale. Questo documento non è un manuale d'istruzione per persone non qualificate. Le istruzioni per il montaggio e l'installazione sono contenute nella Guida di riferimento hardware Twido, TWD USE 10AE.

(c) 2002-2005 Schneider Electric Tutti i diritti riservati

Informazioni aggiuntive sulla sicurezza

Le persone responsabili dell'applicazione, implementazione o uso di questo prodotto devono garantire che in ogni applicazione siano state adottate le opportune misure di progettazione, nel completo rispetto delle leggi applicabili, nonché dei requisiti, delle disposizioni, dei codici e degli standard relativi alle prestazioni e alla sicurezza.

Avvertenze e avvisi generali

⚠ PERICOLO

RISCHIO DI SCARICHE ELETTRICHE, USTIONI O ESPLOSIONE

Disattivare l'alimentazione prima di eseguire operazioni di installazione, rimozione, cablaggio, manutenzione o ispezione del sistema smart relè.

La mancata osservanza di questa precauzione causarà gravi rischi per l'incolumità personale o danni alle apparecchiature.

⚠ AVVERTENZA

RISCHIO DI ESPLOSIONE

- La sostituzione di componenti può pregiudicare l'idoneità dell'apparecchiatura per la classe I, div 2.
- Non scollegare alcuna apparecchiatura a meno che non sia stata tolta l'alimentazione o non si sia accertato che l'area non è soggetta a rischi.

La mancata osservanza di questa precauzione può causare gravi rischi per l'incolumità personale o danni alle apparecchiature.

⚠ AVVERTENZA

FUNZIONAMENTO NON PREVISTO DI APPARECCHIATURE

- Disattivare l'alimentazione prima di eseguire operazioni di installazione, rimozione, cablaggio o manutenzione.
- Questo prodotto non è previsto per l'uso in condizioni di sicurezza critiche. Qualora sussista il rischio di danni personali o alle apparecchiature, utilizzare i necessari interblocchi di sicurezza cablati.
- Non disassemblare, riparare o modificare i moduli.
- Questo controller è progettato per essere utilizzato all'interno di un cabinet.
- Installare i moduli nelle condizioni ambientali operative descritte.
- Utilizzare l'alimentatore sensori solo per alimentare i sensori collegati al modulo.
- Per le linee di alimentazione e i circuiti di uscita, utilizzare un fusibile di tipo T conforme alla norma IEC60127. Il fusibile deve soddisfare i requisiti di tensione e di corrente del circuito. Fusibile raccomandato: Littelfuse® serie 218, 5x20 mm ritardato (ad azione lenta).

La mancata osservanza di questa precauzione può causare gravi rischi per l'incolumità personale o danni alle apparecchiature.

Smaltimento sicuro delle batterie

I controller di base compatti TWDLCA•40DRF utilizzano una batteria al litio esterna opzionale che consente una durata maggiore del backup dei dati. (Nota: la batteria al litio non è fornita con i controller di base compatti ed è necessario acquistarla separatamente).

⚠ AVVERTENZA

RISCHIO DA SOSTANZE TOSSICHE E DA ESPLOSIONE

- Non gettare la batteria al litio nei rifiuti trattati da un inceneritore poiché vi è il rischio che esploda e rilasci sostanze tossiche.
- Non manipolare batterie al litio danneggiate o che presentano delle perdite.
- Le batterie scariche, non più utilizzabili, devono essere smaltite in modo appropriato per evitare rischi alle persone e danni all'ambiente.
- In alcune paesi, è proibito eliminare le batterie al litio insieme ai normali rifiuti domestici o d'ufficio. In ogni caso, l'utente è tenuto a rispettare le normative locali vigenti relative allo smaltimento delle batterie.

La mancata osservanza di questa precauzione può causare gravi rischi per l'incolumità personale o danni alle apparecchiature.

Avvertenza sull'inversione di polarità:

Non è ammessa l'inversione della polarità all'uscita transistor
Le uscite transistor dei controller di base compatti TWDLCA•40DRF non ammettono l'inversione della polarità.

⚠ ATTENZIONE

RISCHIO DI DANNI ALLE USCITE TRANSISTOR PROVOCATE DALL'INVERSIONE DELLA POLARITÀ

- Verificare di aver rispettato le indicazioni di polarità riportate sui morsetti delle uscite transistor.
- L'uso di polarità inversa può provocare danni permanenti o distruggere i circuiti di uscita.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

Informazioni su...

In breve

Scopo del documento

Questa è la Guida di riferimento del software per i controller programmabili Twido ed è costituita dalle seguenti parti principali:

- Descrizione del software di programmazione Twido e introduzione ai concetti fondamentali necessari per programmare i controller Twido.
- Descrizione delle comunicazioni, della gestione degli I/O analogici, dell'implementazione del modulo di interfaccia del bus AS-Interface, del modulo master del bus di campo CANopen e di altre funzioni speciali.
- Descrizione dei linguaggio software utilizzato per creare i programmi Twido.
- Descrizione delle istruzioni e delle funzioni dei controller Twido.

Nota di validità

Le informazioni contenute in questo manuale sono **unicamente** valide per i controller programmabili Twido.

Avvertenza

Schneider Electric non assume responsabilità per eventuali errori contenuti nel presente documento. Nessuna parte di questo documento può essere riprodotta in qualsiasi forma o con qualsiasi mezzo, inclusi quelli elettronici, senza previo consenso scritto di Schneider Electric.

Commenti utente

Inviare eventuali commenti all'indirizzo e-mail TECHCOMM@modicon.com

Descrizione del software Twido

I

In breve

Contenuto della sezione

Questa sezione introduce ai linguaggi software e fornisce le informazioni di base per creare programmi per i controller programmabili Twido.

Contenuto di questa parte

Questa parte contiene i seguenti capitoli:

Capitolo	Titolo del capitolo	Pagina
1	Introduzione al software Twido	19
2	Oggetti del linguaggio Twido	25
3	Memoria utente	53
4	Modalità operative del controller	65
5	Gestione dei task di evento	81

Introduzione al software Twido

1

In breve

Oggetto di questo capitolo

Questo capitolo fornisce una breve introduzione a TwidoSoft, il software di programmazione e configurazione dei controller Twido e ai linguaggi di programmazione List, Ladder e Grafcet utilizzati per creare i programmi di controllo.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Introduzione a TwidoSoft	20
Introduzione ai linguaggi Twido	21

Introduzione a TwidoSoft

Introduzione

TwidoSoft è un ambiente di sviluppo grafico per la creazione, la configurazione e la manutenzione delle applicazioni per i controller programmabili Twido. TwidoSoft consente di creare programmi con diversi tipi di linguaggio (Vedi *Linguaggi Twido, p. 21*), quindi di trasferire l'applicazione per eseguirla in un controller.

TwidoSoft

TwidoSoft è un programma a 32 bit per personal computer (PC) utilizzabile con i sistemi operativi Microsoft Windows 98 Second Edition, Windows 2000 Professional o Microsoft Windows XP .
Le principali funzioni software di TwidoSoft sono:

- Interfaccia utente standard Windows
- Programmazione e configurazione dei controller Twido
- Comunicazioni e controllo del controller

Nota: il collegamento controller-PC utilizza il protocollo TCP/IP. Questo protocollo deve essere quindi installato sul PC.

Configurazione minima

La configurazione minima richiesta per l'utilizzo di TwidoSoft è la seguente:

- Pentium 300MHz,
- 128 MB di RAM,
- 40 MB di spazio disponibile sul disco rigido.

Introduzione ai linguaggi Twido

Introduzione

Un controller programmabile legge degli ingressi, gestisce le uscite e risolve una logica basata su un programma. Per creare un programma di un controller Twido è necessario scrivere una serie di istruzioni in uno dei linguaggi di programmazione Twido.

Linguaggi Twido

Per creare i programmi di controllo Twido, è possibile utilizzare i linguaggi seguenti:

- Linguaggio List:
  Un programma Lista d'istruzioni è costituita da una serie di espressioni logiche scritte come sequenza di istruzioni booleane.
- Linguaggio Ladder:
  Un diagramma Ladder è la rappresentazione grafica di un'espressione logica.
- Linguaggio Grafcet:
  Il linguaggio Grafcet è costituito da una serie di tappe e di transizioni. Twido supporta l'utilizzo delle istruzioni di lista Grafcet, ma non il Grafcet grafico.

È possibile utilizzare un personal computer (PC) per creare e modificare i programmi di controllo Twido tramite questi linguaggi di programmazione.
La funzione di reversibilità List/Ladder consente di invertire un programma da Ladder a List e da List a Ladder.

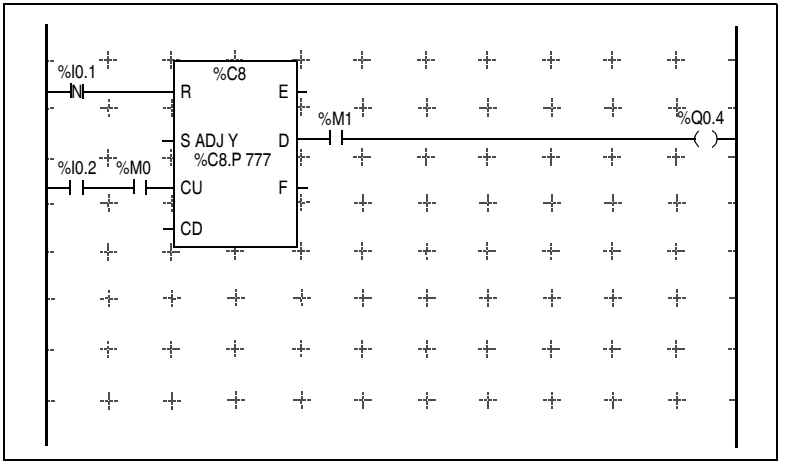
Linguaggio List

Un programma scritto in linguaggio List è costituito da una serie di istruzioni eseguite sequenzialmente dal controller. Di seguito è riportato un esempio di programma List.

```
0    BLK     %C8
1    LDF     %I0.1
2    R
3    LD      %I0.2
4    AND     %M0
5    CU
6    OUT_BLK
7    LD      D
8    AND     %M1
9    ST      %Q0.4
10   END_BLK
```

Linguaggi Ladder

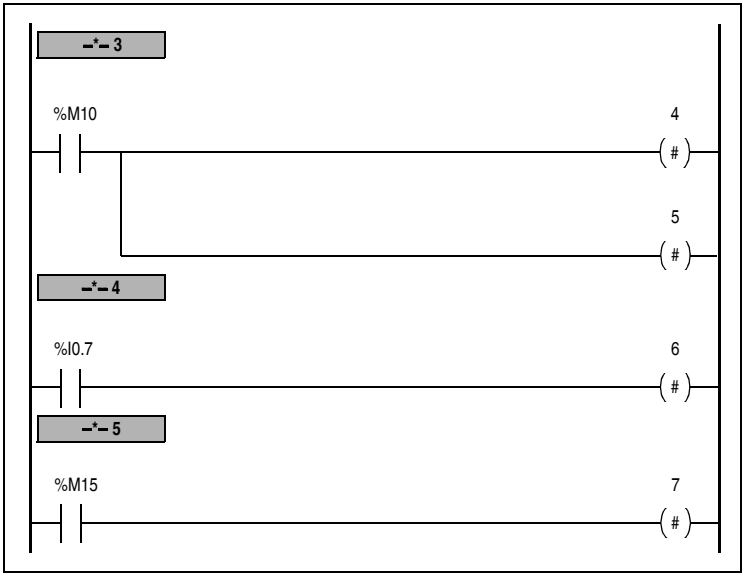
I linguaggi Ladder sono simili ai diagrammi logici di relè che rappresentano i circuiti di controllo dei relè. In questi diagrammi, gli elementi grafici, come ad esempio le bobine, i contatti e i blocchi rappresentano le istruzioni del programma. Di seguito è riportato un esempio di diagramma Ladder.



Linguaggio Grafcet

Il metodo analitico Grafcet divide i sistemi di controllo sequenziali in una serie di passi a cui sono associate azioni, transizioni e condizioni. Di seguito sono riportati alcuni esempio di istruzioni Grafcet che si trovano rispettivamente in programmi List e Ladder.

```
0    -*-    3
1    LD     %M10
2    #      4
3    #      5
4    -*-    4
5    LD     %I0.7
6    #      6
7    -*-    5
8    LD     %M15
9    #      7
10   ...
```



Oggetti del linguaggio Twido

2

In breve

Oggetto di questo capitolo

Questo capitolo fornisce informazioni dettagliate sugli oggetti del linguaggio utilizzato per la programmazione dei controller Twido.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Convalida degli oggetti	26
Oggetti bit	27
Oggetti parola	29
Oggetti a virgola mobile e parola doppia	32
Indirizzamento di oggetti bit	36
Indirizzamento di oggetti parola	37
Indirizzamento di oggetti a virgola mobile	38
Indirizzamento di oggetti parole doppie	39
Indirizzamento degli I/O	40
Indirizzamento di rete	43
Oggetti blocco funzione	44
Oggetti strutturati	46
Oggetti indicizzati	50
Oggetti di simbolizzazione	52

Convalida degli oggetti

Introduzione

Gli oggetti parola e bit sono validi se ad essi è stato assegnato spazio di memoria nel controller. A tale scopo, devono essere utilizzati nell'applicazione prima dello scaricamento sul controller.

Esempio

L'intervallo di oggetti validi è compreso tra zero e il riferimento massimo per il tipo di oggetto in questione. Ad esempio, se il riferimento massimo dell'applicazione per le parole di memoria è %MW9, viene assegnato spazio di memoria alle parole da %MW0 a %MW9. %MW10 in questo esempio non è valido e non è possibile accedervi né internamente né esternamente.

Oggetti bit

Introduzione

Gli oggetti bit sono variabili software di tipo bit che possono essere utilizzate come operandi e provati dalle istruzioni booleane. Di seguito viene fornito un elenco di oggetti bit:

- bit di I/O
- Bit interni (bit di memoria)
- Bit di sistema
- Bit di passo
- Bit estratti da parole

Elenco di bit operandi

La tabella seguente elenca e descrive tutti gli oggetti bit principali utilizzati come operandi nelle istruzioni booleane.

Tipo	Descrizione	Indirizzo o valore	Numero massimo	Accesso in scrittura (1)
Valori immediati	0 o 1 (Falso o Vero)	0 o 1	-	-
Ingressi Uscite	Questi bit sono l'"immagine logica" degli stati elettrici dell'I/O. Sono archiviati nella memoria dati e aggiornati durante ogni scansione della logica di programma.	%Ix.y.z (2) %Qx.y.z (2)	Nota (4)	No Sì
AS-Interface Ingressi Uscite	Questi bit sono l'"immagine logica" degli stati elettrici dell'I/O. Sono archiviati nella memoria dati e aggiornati durante ogni scansione della logica di programma.	 %IAx.y.z %QAx.y.z	Nota (5)	 No Sì
Interna (Memoria)	I bit interni sono aree di memoria interna utilizzate per memorizzare valori intermedi durante l'esecuzione di un programma. **Nota:** i bit di I/O inutilizzati non possono essere usati come bit interni.	%Mi	128 TWDLC•A10DRF, TWDLC•A16DRF 256 Tutti gli altri controller	Sì
Sistema	I bit di sistema da %S0 a %S127 controllano il corretto funzionamento del controller e l'esecuzione corretta del programma applicativo.	%Si	128	In base a i
Blocchi funzione	I bit del blocco funzione corrispondono alle uscite dei blocchi funzione. Queste uscite possono essere collegate direttamente o utilizzate come oggetti.	%TMi.Q, %Ci.P, ecc.	Nota (4)	No (3)
Blocchi funzioni reversibili	Blocchi funzione programmati utilizzando istruzioni di programmazione reversibili BLK, OUT_BLK e END_BLK.	E, D, F, Q, TH0, TH1	Nota (4)	No

Tipo	Descrizione	Indirizzo o valore	Numero massimo	Accesso in scrittura (1)
Estratti di parole	Uno dei 16 bit in alcune parole può essere estratto come bit di operando.	Variabile	Variabile	Variabile
Passi di Grafcet	I bit da %X1 a %Xi sono associati ai passi di Grafcet. Il bit di passo Xi è impostato a 1 quando il passo corrispondente è attivo e impostato a 0 quando il passo è disattivato.	%X21	62 TWDLC•A10DRF, TWDLC•A16 DRF Controller 96 TWDLC•A24DRF, TWDLCA•40DRF e modulari	Sì

Legende:

1. Scritto con il programma o tramite l'editor delle tabelle di animazione.
2. Vedere Indirizzamento di I/O.
3. Ad eccezione di %SBRi.j e %SCi.j, questi bit possono essere letti e scritti.
4. Il numero viene determinato dal modello del controller.
5. Dove x = indirizzo del modulo di espansione (0..7); y = indirizzo AS-Interface (0A..31B); z = numero di canale (0..3). (Vedere *Indirizzamento degli ingressi/uscite associati alle apparecchiature slave collegate sul bus AS-Interface V2, p. 232*.)

Oggetti parola

Introduzione

Gli oggetti parola sono indirizzati in forma di parole a 16 bit contenute nella memoria dati e possono contenere un valore intero compreso tra –32.768 e 32.767 (ad eccezione della funzione di contatore veloce che è compresa tra 0 e 65.535).
Esempi di oggetti parola:

- Valori immediati
- Parole interne (%MWi) (parole di memoria)
- Costanti (%KWi)
- Parole di scambio I/O (%IWi, %QWi%)
- Parole I/O analogici AS-Interface (IWAi, %QWAi)
- Parole di sistema (%SWi)
- Blocchi funzione (dati di configurazione e/o di runtime)

Formati delle parole

Il contenuto delle parole o dei valori è conservato nella memoria utente in codice binario a 16 bit (complemento di due) utilizzando la seguente convenzione:

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	Posizione dei bit
	0	1	0	1	0	0	1	0	0	1	0	0	1	1	0	1	Stato del bit
|+	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1		Valore del bit

Nella notazione binaria con segno, il bit 15 è assegnato per convenzione al segno del valore codificato:

- Il bit 15 è = 0: il contenuto della parola è un valore positivo.
- Il bit 15 è = 1: il contenuto della parola è un valore negativo (i valori negativi sono espressi nella logica del complemento di due).

Le parole e i valori immediati possono essere immessi o richiamati nei seguenti formati:

- Decimale
  Min: -32.768, Max: 32.767 (ad esempio 1579)
- Esadecimale
  Min: 16#0000, Max: 16#FFFF (ad esempio, 16#A536)
  Sintassi alternativa: #A536

Descrizione degli oggetti parola La tabella seguente descrive gli oggetti parola.

Parole	Descrizione	Indirizzo o valore	Numero massimo	Accesso in scrittura (1)
Valori immediati	Questi sono valori interi nello stesso formato delle parole a 16 bit, che abilitano i valori da assegnare a tali parole.		-	No
	Base 10	da -32768 a 32767.		
	Base 16	da 16#0000 a 16#FFFF		
Interna (Memoria)	Utilizzate come parole di "lavoro" per memorizzare i valori durante il funzionamento nella memoria dati. Le parole da %MW0 a %MW255 sono lette e scritte direttamente dal programma.	%MWi	3000	Sì
Costanti	Memorizzano costanti o messaggi alfanumerici. Il loro contenuto può essere scritto o modificato solo utilizzando TwidoSoft durante la configurazione. Il programma può accedere in lettura solo alle costanti da %KW0 a %KW63.	%KWi	256	Sì, solo utilizzando TwidoSoft
Sistema	Queste parole a 16 bit hanno varie funzioni: ● Fornire accesso ai dati che provengono direttamente dal controller leggendo le parole %SWi. ● Eseguire operazioni sull'applicazione (ad esempio, regolazione dei blocchi orologio-datario).	%SWi	128	In base a i
Blocchi funzione	Queste parole corrispondono ai parametri o ai valori correnti dei blocchi funzione.	%TM2.P, %Ci.P, ecc.		Sì
Parole di scambio di rete	Assegnate ai controller collegati come collegamenti remoti. Queste parole vengono utilizzate per la comunicazione tra i controller:			
	Ingresso di rete	%INWi.j	4 per collegamento remoto	No
	Uscita di rete	%QNWi.j	4 per collegamento remoto	Sì
Parole I/O analogici	Assegnate a ingressi e uscite analogici di moduli slave AS-Interface.			
	Ingressi analogici	%IWAx.y.z	Nota (3)	No
	Uscite analogiche	%QWAx.y.z	Nota (3)	Sì

Parole	Descrizione	Indirizzo o valore	Numero massimo	Accesso in scrittura (1)
Bit estratti	È possibile estrarre uno dei 16 bit dalle seguenti parole:			
	Interna	%MWi:Xk	1500	Sì
	Sistema	%SWi:Xk	128	In base a i
	Costanti	%KWi:Xk	64	No
	Ingresso	%IWi.j:Xk	Nota (2)	No
	Uscita	%QWi.j:Xk	Nota (2)	Sì
	Ingresso slave AS-Interface	%IWAx.y.z:Xk	Nota (2)	No
	Uscita slave AS-Interface	%QWAx.y.z:Xk	Nota (2)	Sì
	Ingresso di rete	%INWi.j:Xk	Nota (2)	No
	Uscita di rete	%QNWi.j:Xk	Nota (2)	Sì

Nota:
1. Scritto con il programma o tramite l'editor delle tabelle di animazione.
2. Il numero viene determinato dalla configurazione.
3. Dove x = indirizzo del modulo di espansione (0..7); y = indirizzo AS-Interface (0A..31B); z = numero di canale (0..3). (Vedere *Indirizzamento degli ingressi/uscite associati alle apparecchiature slave collegate sul bus AS-Interface V2, p. 232*.)

Oggetti a virgola mobile e parola doppia

Introduzione

Il programma TwidoSoft consente di eseguire operazioni sugli oggetti a virgola mobile e a parola doppia intera.
La virgola mobile è un argomento matematico che presenta un decimale nell'espressione (esempi: 3,4E+38; 2,3 o 1,0).
Un parola doppia intera è costituita da 4 byte memorizzati nella memoria dati che possono contenere un valore intero compreso tra -2147483648 e +2147483647.

Formato e valore della virgola mobile

Il formato a virgola mobile corrisponde alla norma IEEE STD 734-1985 (equivalente a IEC 559). La lunghezza delle parole è di 32 bit, che corrisponde a numeri decimali a precisione semplice.
Tabella di rappresentazione del formato in virgola mobile:

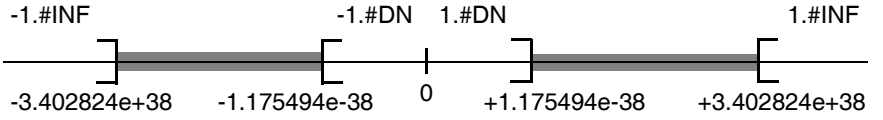
Bit 31	Bit {30...23}	Bit {22...0}
S	Esponente	Mantissa

Il valore del formato precedente viene determinato dall'equazione seguente:

Valore in virgola mobile a 32-bit = $(-1)^{S} * 2^{(\text{Exposant} - 127)} * 1.\text{Mantissa}$

I valori in virgola mobile possono essere rappresentati con o senza esponenti e devono sempre includere una virgola (virgola mobile).
I valori in virgola mobile sono compresi tra -3.402824e+38 e -1.175494e-38; 1.175494e-38 e 3.402824e+38 (valori in grigio sullo schema). Includono anche il valore 0 con notazione 0.0

.



Quando un risultato di calcolo è:
- inferiore a -3.402824e+38, viene visualizzato il simbolo -1.#INF (per –infinito)
- superiore a +3.402824e+38, viene visualizzato il simbolo 1.#INF (per +infinito)
- compreso tra 1,175494e-38 e 1,175494e-38, viene arrotondato a 0.0. Un valore compreso tra questi limiti non può essere immesso come valore in virgola mobile.
- non definito (ad esempio radice quadrata di un numero negativo) viene visualizzato il simbolo 1.#NAN o -1.#NAN.

La precisione di rappresentazione è 2-24. Per visualizzare numeri in virgola mobile non è necessario visualizzare più di 6 cifre dopo la virgola decimale.

Nota:
- Il valore "1285" viene interpretato come valore intero; per poter essere considerato come valore in virgola mobile deve essere scritto: "1285.0"

Campo limite delle funzioni aritmetiche in virgola mobile

La seguente tabella descrive il campo limite delle funzioni aritmetiche sugli oggetti a virgola mobile

Funzione aritmetica		**Campo limite ed operazioni non valide**	
Tipo	**Sintassi**	**#QNAN (non valido)**	**#INF (infinito)**
Radice quadrata di un operando	SQRT(x)	x < 0	x > 1,7E38
Potenza di un valore intero per un numero reale EXPT(%MF,%MW)	EXPT(y, x) (dove: x^y = %MW^%MF)	x < 0	y.ln(x) > 88
Logaritmo in base 10	LOG(x)	x <= 0	x > 2,4E38
Logaritmo naturale	LN(x)	x <= 0	x > 1,65E38
Esponenziale naturale	EXP(x)	x < 0	x > 88.0

Compatibilità hardware

Le operazioni in virgola mobile e con parole doppie non sono supportate da tutti i controller Twido.
La tabella che segue descrive la compatibilità hardware:

Controller Twido	**Parole doppie supportate**	**Virgole mobili supportate**
TWDLMDA40DUK	Sì	Sì
TWDLMDA40DTK	Sì	Sì
TWDLMDA20DUK	Sì	No
TWDLMDA20DTK	Sì	No
TWDLMDA20DRT	Sì	Sì
TWDLCA•40DRF	Sì	Sì
TWDLC•A24DRF	Sì	No
TWDLC•A16DRF	Sì	No
TWDLC•A10DRF	No	No

Controllo validità

Il bit sistema %S18 viene impostato a 1 quando il risultato non rientra nei limiti validi. I bit della parola di stato %SW17 indicano la causa di un errore in un'operazione a virgola mobile:
Bit differenti della parola %SW17:

%SW17:X0	operazione non valida, il risultato non è un numero (1.#NAN o -1.#NAN)
%SW17:X1	riservato
%SW17:X2	divisione per 0, il risultato è infinito (-1.#INF o 1.#INF)
%SW17:X3	risultato superiore in valore assoluto a +3.402824e+38, il risultato è infinito (-1.#INF o 1.#INF)
da %SW17:X4 a X15	riservato

Questa parola viene azzerata dal sistema in seguito ad avvio a freddo e dal programma per il riutilizzo.

Descrizione degli oggetti a virgola mobile e parola doppia

La tabella seguente descrive gli oggetti a virgola mobile e parola doppia:

Tipo di oggetti	Descrizione	Indirizzi	Numero massimo	Accesso in scrittura	Forma indicizzata
Valori immediati	Interi o decimali con formato identico a quello degli oggetti a 32 bit.	-	[-]	No	-
virgola mobile interna	Oggetti utilizzati per memorizzare i valori durante il funzionamento nella memoria dati.	%MFi	1500	Sì	%MFi[indice]
parola doppia interna		%MDi	1500	Sì	%MDi[indice]
costante in virgola mobile	memorizzano le costanti.	%KFi	128	Sì, solo utilizzando TwidoSoft	%KFi[indice]
costante doppia		%KDi	128	Sì, solo utilizzando TwidoSoft	%KDi[indice]

Possibilità di sovrapposizione tra oggetti

Le parole a lunghezza singola, doppia e in virgola mobile sono ordinate all'interno dello spazio dati di una stessa area di memoria. La parola in virgola mobile %MFi e la parola doppia %MDi corrispondono alle parole a lunghezza singola %MWi e %MWi+1 (la parola %MWi racchiude i bit meno significativi e la parola %MWi+1 i bit più significativi della parola %MFi).
La tabella seguente descrive la sovrapposizione di virgole mobili e parole doppie interne:

Virgola mobile e doppia	indirizzo dispari	Parole interne
%MF0 / %MD0		%MW0
	%MF1 / %MD1	%MW1
%MF2 / %MD2		%MW2
	%MF3 / %MD3	%MW3
%MF4 / %MD4		%MW4
	...	%MW5
...		...
	%MFi / %MDi	%MWi
%MFi+1 / %MDi+1		%MWi+1

La tabella seguente descrive la sovrapposizione di valori in virgola mobile e costanti doppie:

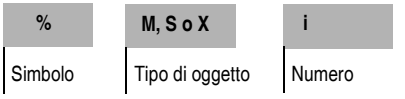
Virgola mobile e costanti doppie	Indirizzo dispari	Parole interne
%KF0 / %KD0		%KW0
	%KF1 / %KD1	%KW1
%KF2 / %KD2		%KW2
	%KF3 / %KD3	%KW3
%KF4 / %KD4		%KW4
	...	%KW5
...		...
	%kFi / %kDi	%KWi
%KFi+1 / %KDi+1		%KWi+1

Esempio:
%MF0 corrisponde a %MW0 e %MW1. %KF543 corrisponde a %KW543 e %KW544.

Indirizzamento di oggetti bit

Sintassi

Utilizzare il seguente formato per indirizzare le parole interne, le parole di sistema e gli oggetti bit dei passi:



Descrizione

La tabella seguente descrive gli elementi del formato di indirizzamento.

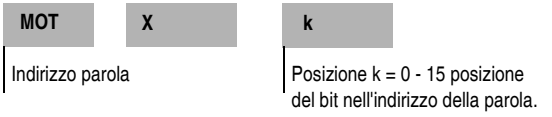
Gruppo	Elemento	Descrizione
Simbolo	%	Il simbolo di percentuale precede sempre una variabile software.
Tipo di oggetto	M	I bit interni memorizzano valori intermedi durante l'esecuzione di un programma.
	S	I bit di sistema indicano lo stato e informazioni di controllo per il controller.
	X	I bit di passo forniscono lo stato delle attività del passo.
Numero	i	Il valore numerico massimo dipende dal numero di oggetti configurati.

Esempi di indirizzamento di oggetti bit:

- %M25 = bit interno numero 25
- %S20 = bit di sistema numero 20
- %X6 = bit di passo numero 6

Oggetti bit estratti dalle parole

TwidoSoft viene utilizzato per estrarre uno dei 16 bit dalle parole. L'indirizzo della parola viene quindi completato dalla riga di bit estratta in base alla seguente sintassi:



Esempi:

- %MW5:X6 = bit numero 6 della parola interna %MW5
- %QW5.1:X10 = bit numero 10 della parola di uscita %QW5.1

Indirizzamento di oggetti parola

Introduzione

L'indirizzamento degli oggetti parola avviene seguendo la sintassi descritta di seguito. Questa sintassi non si applica all'indirizzamento di I/O (vedere la sezione *Indirizzamento degli I/O, p. 40*) e dei blocchi funzione (vedere la sezione *Oggetti blocco funzione, p. 44*).

Sintassi

Utilizzare il seguente formato per indirizzare le parole interne, le costanti e le parole di sistema:



Descrizione

La tabella seguente descrive gli elementi del formato di indirizzamento.

Gruppo	Elemento	Descrizione
Simbolo	%	Il simbolo di percentuale precede sempre un indirizzo interno.
Tipo di oggetto	M	Le parole interne memorizzano valori intermedi durante l'esecuzione di un programma.
	K	Le costanti memorizzano valori costanti o messaggi alfanumerici. Il loro contenuto può essere scritto o modificato solo utilizzando TwidoSoft.
	S	Le parole di sistema indicano lo stato e informazioni di controllo per il controller.
Sintassi	W	parola a 16 bit.
Numero	i	Il valore numerico massimo dipende dal numero di oggetti configurati.

Esempi di indirizzamento di oggetti parola:

- %MW15 = parola interna numero 15
- %KW26 = costante numero 26
- %SW30 = parola di sistema numero 30

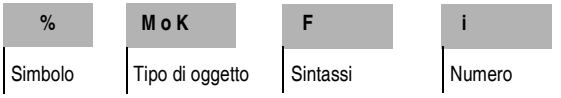
Indirizzamento di oggetti a virgola mobile

Introduzione

L'indirizzamento degli oggetti a virgola mobile avviene seguendo la sintassi descritta di seguito. Questa sintassi non si applica all'indirizzamento di I/O (vedere la sezione *Indirizzamento degli I/O, p. 40*) e dei blocchi funzione (vedere la sezione *Oggetti blocco funzione, p. 44*).

Sintassi

Utilizzare il seguente formato per indirizzare gli oggetti a virgola mobile interni e le costanti:



Descrizione

La tabella seguente descrive gli elementi del formato di indirizzamento.

Gruppo	Elemento	Descrizione
Simbolo	%	Il simbolo di percentuale precede sempre un indirizzo interno.
Tipo di oggetto	M	Gli oggetti a virgola mobile interni memorizzano valori intermedi durante l'esecuzione di un programma.
	K	Gli oggetti a virgola mobile costanti memorizzano valori costanti. Il loro contenuto può essere scritto o modificato solo utilizzando TwidoSoft.
Sintassi	F	Oggetto a 32 bit.
Numero	i	Il valore numerico massimo dipende dal numero di oggetti configurati.

Esempi di indirizzamento di oggetti a virgola mobile:

- %MF15 = oggetto a virgola mobile interno numero 15
- %KF26 = oggetto a virgola mobile costante numero 26

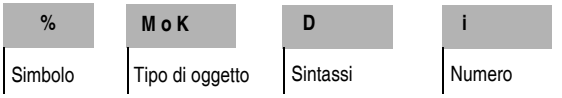
Indirizzamento di oggetti parole doppie

Introduzione

L'indirizzamento degli oggetti parole doppie avviene seguendo la sintassi descritta di seguito. Questa sintassi non si applica all'indirizzamento di I/O (vedere la sezione *Indirizzamento degli I/O, p. 40*) e dei blocchi funzione (vedere la sezione *Oggetti blocco funzione, p. 44*).

Sintassi

Utilizzare il seguente formato per indirizzare le parole doppie interne e le costanti:



Descrizione

La tabella seguente descrive gli elementi del formato di indirizzamento.

Gruppo	Elemento	Descrizione
Simbolo	%	Il simbolo di percentuale precede sempre un indirizzo interno.
Tipo di oggetto	M	Le parole doppie interne consentono di memorizzare valori intermedi durante l'esecuzione di un programma.
	K	Le parole doppie costanti permettono di memorizzare valori costanti o messaggi alfanumerici. Il loro contenuto può essere scritto o modificato solo utilizzando TwidoSoft.
Sintassi	D	Parola doppia di 32 bit.
Numero	i	Il valore numerico massimo dipende dal numero di oggetti configurati.

Esempi di indirizzamento di oggetti parole doppie:

- %MD15 = parola doppia interna numero 15
- %KD26 = parola doppia costante numero 26

Indirizzamento degli I/O

Introduzione

Ciascun punto di ingresso/uscita (I/O) in una configurazione Twido ha un indirizzo univoco: L'indirizzo "%I0.0.4", ad esempio, rappresenta solo l'ingresso 4 di un controller.

È possibile assegnare gli indirizzi I/O ai seguenti dispositivi hardware:

- Controller configurato come master di collegamento remoto
- Controller configurato come I/O remoto
- Moduli di espansione degli I/O

Il modulo di interfaccia del bus AS-Interface di TWDNOI10M3 e il modulo del bus di campo CANopen di TWDNCO1M utilizzano ognuno un sistema di indirizzi proprio speciale per l'indirizzamento dei dispositivi slave di I/O collegati al rispettivo bus:

- Per TWDNOI10M3, vedere *Indirizzamento degli ingressi/uscite associati alle apparecchiature slave collegate sul bus AS-Interface V2, p. 232*.
- Per TWDNCO1M, vedere *Indirizzamento dei PDO del master CANopen, p. 273*.

Riferimenti multipli a un'uscita o bobina

In un programma, è possibile avere più riferimenti per una singola uscita o bobina. Sulle uscite hardware viene aggiornato solo il risultato dell'ultimo aggiornamento. Ad esempio, %Q0.0.0 può essere utilizzato più di una volta in un programma e non verranno emessi avvisi per le occorrenze multiple. È quindi importante convalidare solo l'equazione che fornirà lo stato dell'uscita desiderato.

⚠ ATTENZIONE

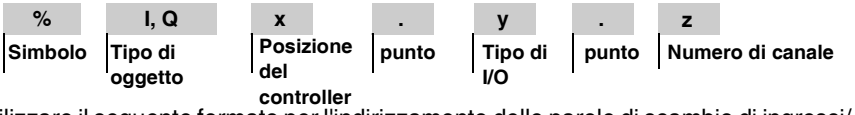
FUNZIONAMENTO IMPREVISTO

Non sono forniti doppi controlli delle uscite o avvertenze. Rivedere l'uso delle uscite o delle bobine prima di modificarle nell'applicazione.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

Formato

Per l'indirizzamento di ingressi/uscite utilizzare il seguente formato.



Utilizzare il seguente formato per l'indirizzamento delle parole di scambio di ingressi/uscite.



Descrizione

La tabella che segue descrive il formato di indirizzamento di I/O.

Gruppo	Elemento	Valore	Descrizione
Simbolo	%	-	Il simbolo di percentuale precede sempre un indirizzo interno.
Tipo di oggetto	I	-	Ingresso. "Immagine logica" dello stato elettrico dell'ingresso di un controller o di un modulo I/O di espansione.
	Q	-	Uscita. "Immagine logica" dello stato elettrico dell'uscita di un controller o di un modulo I/O di espansione.
Posizione del controller.	x	0 1 - 7	Controller master (master di collegamento remoto). Controller remoto (slave di collegamento remoto).
Tipo di I/O	y	0 1 - 7	I/O della base (I/O locale sul controller). Moduli di I/O di espansione.
Numero di canale	z	0 - 31	Numero del canale di I/O sul controller o modulo di I/O d'espansione. Il numero di punti di I/O disponibili dipende dal modello del controller o dal tipo di modulo di I/O d'espansione.

Esempi

La tabella che segue mostra alcuni esempi di indirizzamento di I/O.

Oggetto di I/O	Descrizione
%I0.0.5	Punto d'ingresso numero 5 sul controller di base (I/O locale).
%Q0.3.4	Punto di uscita n° 4 sul modulo di I/O di espansione dell'indirizzo 3 per il controller di base (I/O di espansione).
%I0.0.3	Punto d'ingresso numero 3 sul controller di base.
%I3.0.1	Punto d'ingresso numero 1 sul controller di I/O remoto all'indirizzo di collegamento remoto numero 3.
%I0.3.2	Punto d'ingresso numero 2 sul modulo di I/O d'espansione all'indirizzo numero 3 per il controller di base.

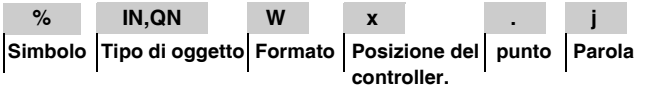
Indirizzamento di rete

Introduzione

I dati dell'applicazione vengono scambiati tra controller peer e il controller master su una rete a collegamento remoto Twido utilizzando le parole di rete %INW e %QNW. Per ulteriori informazioni, consultare *Comunicazioni , p. 87*.

Formato

Per l'indirizzamento di rete utilizzare il seguente formato.



Descrizione del formato

La tabella che segue descrive il formato di indirizzamento di rete.

Gruppo	Elemento	Valore	Descrizione
Simbolo	%	-	Il simbolo di percentuale precede sempre un indirizzo interno.
Tipo di oggetto	IN	-	Parola d'ingresso di rete. Trasferimento dati da master a peer.
	QN	-	Parola d'uscita di rete. Trasferimento dati da peer a master.
Formato	W	-	Parola a 16 bit.
Posizione del controller.	x	0 1 - 7	Controller master (master a collegamento remoto). Controller remoto (slave a collegamento remoto).
Parola	j	0 - 3	Ciascun controller peer utilizza una-quattro parole per scambiare i dati con il controller master.

Esempi

La tabella che segue mostra alcuni esempi di indirizzamento di rete.

Oggetto di rete	Descrizione
%INW3.1	Parola di rete numero 1 del controller remoto numero 3.
%QNW0.3	Parola di rete numero 3 del controller della base.

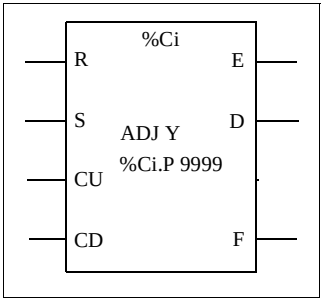
Oggetti blocco funzione

Introduzione

I blocchi funzione forniscono oggetti bit e parole specifiche ai quali è possibile accedere dal programma.

Esempio di un blocco funzione

L'illustrazione seguente mostra un blocco funzione Contatore.



Blocco contatore avanti/indietro

Oggetti bit

Gli oggetti bit corrispondono alle uscite del blocco: è possibile accedere a questi bit tramite le istruzioni di test booleane utilizzando uno dei metodi seguenti:

- Direttamente (ad esempio, LD E) se sono collegati al blocco nella programmazione reversibile (vedere *Principi di programmazione dei blocchi funzione standard, p. 395*).
- Specificando il tipo di blocco (ad esempio, LD %Ci.E).

È possibile accedere agli ingressi sotto forma di istruzioni.

Oggetti parola

Gli oggetti parola corrispondono a parametri e valori specificati, come indicato di seguito:

- Parametri di configurazione blocco: alcuni parametri sono accessibili da parte del programma (ad esempio, parametri di preselezione) e altri non lo sono (ad esempio, base tempo).
- Valori correnti: ad esempio, %Ci.V, il valore del conteggio corrente.

Oggetti parola

Gli oggetti parola doppia aumentano la capacità di calcolo del controller durante l'esecuzione di funzioni di sistema, come contatori veloci (%FC), contatori molto veloci (%VFC) e generatori di impulsi (%PLS).
L'indirizzamento di oggetti parola doppia a 32 bit usato con blocchi funzione prevede semplicemente l'aggiunta del carattere "D" alla sintassi originale degli oggetti parola standard. L'esempio seguente mostra come indirizzare il valore corrente di un contatore veloce in formato standard e in formato parola doppia:

- %FCi.V è il valore corrente del contatore veloce in formato standard.
- %FCi.VD è il valore corrente del contatore veloce in formato parola doppia.

Nota: Gli oggetti parola doppia non sono supportati da tutti i controller Twido. Consultare *Compatibilità hardware, p. 33* per sapere se il proprio controller Twido supporta le parole doppie.

Oggetti accessibili dal programma

Vedere le sezioni seguenti appropriate per un elenco di oggetti accessibili dal programma.

- Per i blocchi funzione di base, consultare *Blocchi funzione di base, p. 393*.
- Per i blocchi funzione avanzati, consultare *Oggetti bit e parola associati ai blocchi funzione avanzati, p. 444*.

Oggetti strutturati

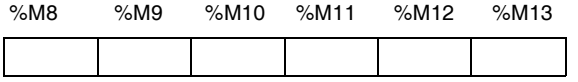
Introduzione

Gli oggetti strutturati sono combinazioni di oggetti adiacenti. Twido supporta i seguenti tipi di oggetti strutturati:

- Stringhe di bit
- Tabelle di parole
- Tabelle di parole doppie
- Tabelle di parole a virgola mobile

Stringhe di bit

Le stringhe di bit sono composte da una serie di bit oggetto adiacenti dello stesso tipo e di lunghezza (L) definita.
Esempio: Stringa di bit %M8:6



Nota: %M8:6 è accettabile (8 è un multiplo di 8), mentre %M10:16 è inaccettabile (10 non è un multiplo di 8).

Le stringhe di bit possono essere utilizzate con l'istruzione di assegnazione (consultare *Istruzioni di assegnazione, p. 417*).

Tipi di bit disponibili

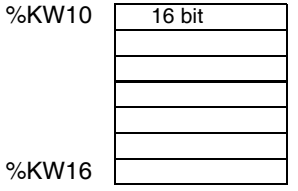
Tipi di bit disponibili per le stringhe di bit:

Tipo	Indirizzi	Dimensione massima	Accesso in scrittura
Bit d'ingresso digitale	%I0.0:L o %I1.0:L (1)	0<L<17	No
Bit d'uscita digitale	%Q0.0:L o %Q1.0:L (1)	0<L<17	Sì
Bit di sistema	%Si:L dove "i" è multiplo di 8	0<L<17 e i+L≤ 128	In base a i
Bit di passi Grafcet	%Xi:L dove "i" è multiplo di 8	0<L<17 e i+L≤ 95 (2)	Sì (da programma)
Bit interni	%Mi:L dove "i" è multiplo di 8	0<L<17 e i+L≤ 256 (3)	Sì

Legenda :
1. Solo i bit di I/O da 0 a 16 possono essere letti in stringa di bit. Per i controller con 24 ingressi e 32 moduli di I/O, i bit oltre 16 non possono essere letti in stringa di bit.
2. Il massimo i+L per TWWDLCAA10DRF e TWDLCAA16DRF è 62
3. Il massimo i+L per TWWDLCAA10DRF e TWDLCAA16DRF è 128

Tabelle di parole

Le tabelle di parole sono composte da una serie di oggetti adiacenti dello stesso tipo e di lunghezza (L) definita.
Esempio: Tabella di parole %KW10:7



Le tabelle di parole possono essere utilizzate con l'istruzione di assegnazione (vedere *Istruzioni di assegnazione, p. 417*).

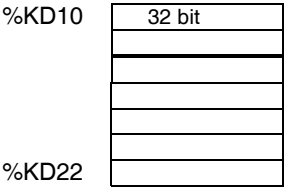
Tipi di parole disponibili

Tipi di parole disponibili per le tabelle di parole:

Tipo	Indirizzi	Dimensione massima	Accesso in scrittura
Parole interne	%MWi:L	0<L<256 e i+L< 3000	Sì
Costanti	%KWi:L	0<L<256 e i+L< 256	No
Parole di sistema	%SWi:L	0<L e i+L<128	In base a i

Tabelle di parole doppie

Le tabelle di parole doppie sono composte da una serie di oggetti adiacenti dello stesso tipo e di lunghezza (L) definita.
Esempio: Tabella di parole doppie %KD10:7



Le tabelle di parole doppie possono essere utilizzate con l'istruzione di assegnazione (vedere *Istruzioni di assegnazione, p. 417*).

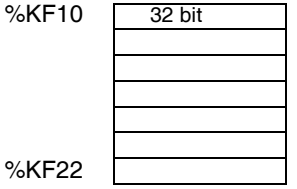
Tipi di parole doppie disponibili

Tipi di parole disponibili per le tabelle di parole doppie:

Tipo	Indirizzi	Dimensione massima	Accesso in scrittura
Parole interne	%MDi:L	0<L<256 e i+L< 3000	Sì
Costanti	%KDi:L	0<L e i+L< 256	No

Tabelle di parole a virgola mobile

Le tabelle di parole a virgola mobile sono composte da una serie di oggetti adiacenti dello stesso tipo e di lunghezza (L) definita.
Esempio: Tabella di parole a virgola mobile %KF10:7



Le tabelle di parole a virgola mobile possono essere utilizzate con l'istruzione di assegnazione (vedere Istruzioni avanzate).

Tipi di parole a virgola mobile disponibili

Tipi di parole disponibili per le tabelle di parole a virgola mobile:

Tipo	Indirizzo	Dimensione massima	Accesso in scrittura
Parole interne	%MFi:L	0<L<256 e i+L< 3000	Sì
Parole costanti	%KFi:L	0<L e i+L<256	No

Oggetti indicizzati

Introduzione

Una parola indicizzata è una parola singola o doppia o una virgola mobile con un indirizzo di oggetto indicizzato. Esistono due tipi di indirizzamento degli oggetti:

- Indirizzamento diretto
- Indirizzamento indicizzato

Indirizzamento diretto

L'indirizzo diretto di un oggetto viene impostato e definito durante la scrittura di un programma.
Esempio: %M26 è un bit interno con indirizzo diretto 26.

Indirizzamento indicizzato

L'indirizzo indicizzato di un oggetto fornisce un metodo per modificare l'indirizzo di un oggetto aggiungendo un indice all'indirizzo diretto dell'oggetto stesso. Il contenuto dell'indice viene aggiunto all'indirizzo diretto dell'oggetto. L'indice viene definito da una parola interna %MWi. Il numero di "parole d'indice" è illimitato.
Esempio: %MW108[%MW2] è una parola con un indirizzo costituito dall'indirizzo diretto 108 più il contenuto della parola %MW2.
Se la parola %MW2 ha valore 12, la scrittura in %MW108[%MW2] equivale alla scrittura in %MW120 (108 + 12).

Oggetti disponibili per l'indirizzamento indicizzato

Di seguito sono indicati i tipi di oggetti per l'indirizzamento indicizzato.

Tipo	Indirizzi	Dimensione massima	Accesso in scrittura
Parole interne	%MWi[MWj]	0≤ i+%MWj<3000	Sì
Costanti	%KWi[%MWj]	0≤ i+%MWj<256	No
Parole doppie interne	%MDi[MWj]	0≤ i+%MWj<2999	Sì
Parole doppie costanti	%KDi[%MWj]	0≤ i+%MWj<255	No
Virgola mobile interna	%MFi[MWj]	0≤ i+%MWj<2999	Sì
Costanti in virgola mobile	%KFi[%MWj]	0≤ i+%MWj<255	No

Gli oggetti indicizzati possono essere utilizzati con le istruzioni di assegnazione (vedere *Istruzioni di assegnazione, p. 417* per parole e parole doppie) e nelle istruzioni di confronto (vedere *Istruzioni di confronto, p. 422* per parole e parole doppie). Questo tipo di indirizzamento consente la scansione in successione di una serie di oggetti dello stesso tipo (ad esempio parole interne e costanti), ad esempio modificando il contenuto dell'oggetto indicizzato tramite il programma.

Bit di sistema di overflow d'indice %S20

L'overflow dell'indice si verifica quando l'indirizzo di un oggetto indicizzato supera i limiti della zona di memoria contenente lo stesso tipo di oggetto. In breve:

- L'indirizzo dell'oggetto più il contenuto dell'indice sono inferiori a 0.
- L'indirizzo dell'oggetto più il contenuto dell'indice sono maggiori della parola più grande referenziata direttamente nell'applicazione. Il numero massimo è 2999 (per parole %MWi) o 255 (per parole %KWi).

Nel caso di overflow dell'indice, il sistema imposta il bit di sistema %S20 a 1 e all'oggetto viene assegnato un valore di indice 0.

Nota: l'utente è responsabile del monitoraggio dell'overflow. Il bit %S20 deve essere letto dal programma utente per una possibile elaborazione. L'utente deve confermare che è stato azzerato.
%S20 (stato iniziale = 0):

- Nell'overflow dell'indice: impostazione a 1 dal sistema.
- Riconoscimento di overflow: azzeramento da parte dell'utente dopo la modifica dell'indice.

Oggetti di simbolizzazione

Introduzione

Utilizzare i simboli per individuare gli oggetti del linguaggio software Twido per nome o per elementi mnemonici personalizzati. Tramite i simboli è possibile esaminare e analizzare rapidamente la logica di un programma. Lo sviluppo e il testing di un'applicazione risultano quindi molto più semplici.

Esempio

WASH_END, ad esempio, è un simbolo utilizzabile per identificare un blocco funzione timer che rappresenta il termine di un ciclo di lavaggio. Questo tipo di nome rende molto più facile l'interpretazione del suo uso che non l'indirizzo di un programma quale %TM3.

Regole per la definizione dei simboli

Di seguito sono riportate alcune regole per la definizione dei simboli:

- Utilizzare al massimo 32 caratteri.
- Utilizzare lettere (A-Z), numeri (0 -9) o il carattere di sottolineatura (_).
- Il primo carattere deve essere alfabetico o accentato. L'utilizzo del segno di percentuale (%) non è ammesso.
- Non sono ammessi spazi o caratteri speciali.
- Non vi è differenza tra maiuscole e minuscole. Pump1 e PUMP1, ad esempio, sono lo stesso simbolo e possono essere utilizzati solo una volta nell'applicazione.

Simboli di modifica

I simboli sono definiti e associati agli oggetti del linguaggio nell'editor dei simboli. I simboli e i relativi commenti sono memorizzati con l'applicazione sul disco fisso del PC e non sul controller. Pertanto, non possono essere trasferiti al controller insieme all'applicazione.

Memoria utente

3

In breve

Oggetto di questo capitolo

Questo capitolo descrive la struttura e l'uso della memoria utente Twido.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Struttura della memoria utente	54
Backup e ripristino senza cartuccia di backup o memoria estesa	56
Backup e ripristino con una cartuccia di backup da 32 K	58
Uso della cartuccia di memoria estesa da 64K	61

Struttura della memoria utente

Introduzione

La memoria del controller accessibile all'applicazione utente è divisa in due gruppi distinti:
- Valori di bit
- Valori di parole (16 bit) e valori di parole doppie (32 bit)

Memoria di bit

La memoria di bit si trova nella RAM interna integrata nel controller. Contiene la mappa di 128 oggetti bit.

Memoria di parole

La memoria delle parole (16 bit) supporta:
- **Parole dinamiche**: memoria di runtime (memorizzata solo nella RAM).
- **Parole di memoria (%MW) e parole doppie (%MD)**: dati di sistema dinamici e dati di sistema.
- **Programma**: descrittori e codice eseguibile per i task.
- **Dati di configurazione**: costanti, valori iniziali e configurazione di ingressi/uscite.

Tipi di memorizzazione

I controller Twido dispongono di tre tipi di memoria:
- Random Access Memory.
  Memoria volatile interna: contiene parole dinamiche, parole di memoria, dati di programma e di configurazione.
- EEPROM
  EEPROM integrata da 32 KB che consente di eseguire il backup interno di programma e di dati. Protegge il programma dal danneggiamento dovuto a esaurimento della batteria o mancanza di alimentazione superiore a 30 giorni. Contiene un programma e dati di configurazione e fino a 512 parole di memoria. Il programma non viene salvato se si utilizza una cartuccia di memoria estesa da 64 K e se Twido è stato configurato per accettare cartucce di memoria estesa da 64 K.
- Cartuccia di backup da 32 K
  Cartuccia esterna opzionale utilizzata per salvare un programma e per trasferirlo ad altri controller Twido. Può essere utilizzata per aggiornare il programma nella RAM del controller. Contiene programma e costanti ma non parole di memoria.
- Cartuccia di memoria estesa da 64 K
  Cartuccia esterna opzionale che memorizza un programma fino a 64 KB. Deve restare inserita nel controller per tutto il tempo in cui viene utilizzato il programma.

Memoria di salvataggio

È possibile salvare il programma e le parole di memoria del controller:
- nella RAM (fino a 30 giorni con batterie in buone condizioni)
- nella memoria EEPROM (max 32 KB)

Il trasferimento del programma dalla memoria EEPROM alla memoria RAM avviene automaticamente quando si perde il programma contenuto nella RAM (oppure se non vi sono batterie).
Il trasferimento manuale può essere eseguito anche con TwidoSoft.

Configurazioni di memoria

Le tabelle seguenti descrivono i tipi di configurazioni di memoria possibili con i controller Twido compatti e modulari.

Tipo di memoria	Controller compatti				
	10DRF	**16DRF**	**24DRF**	**40DRF (32k)**	**40DRF** (64k)**
RAM interna Mem 1*	10KB	10KB	10KB	10KB	10KB
RAM esterna Mem 2*		16KB	32KB	32KB	64KB
EEPROM interna	8KB	16KB	32KB	32KB	32KB***
EEPROM esterna	32KB	32KB	32KB	32KB	64KB
Dimensione massima del programma	8KB	16KB	32KB	32KB	64KB
Backup esterno massimo	8KB	16KB	32KB	32KB	64KB

Tipo di memoria	Controller modulari		
	20DUK 20DTK	**20DRT 40DUK 40DTK (32k)**	**20DRT 40DUK 40DTK** (64k)**
RAM interna Mem 1*	10KB	10KB	10KB
RAM esterna Mem 2*	32KB	32KB	64KB
EEPROM interna	32KB	32KB	32KB***
EEPROM esterna	32KB	32KB	64KB
Dimensione massima del programma	32KB	32KB	64KB
Backup esterno massimo	32KB	32KB	64KB

(*) Mem 1 e Mem 2 del bilancio memoria.
(**) in questo caso, la cartuccia da 64KB deve rimanere installata in Twido e dichiarata nella configurazione, se non lo è.
(***) riservati per il backup delle prime 512 parole %MW o delle prime 256 parole doppie %MD.

Backup e ripristino senza cartuccia di backup o memoria estesa

Introduzione

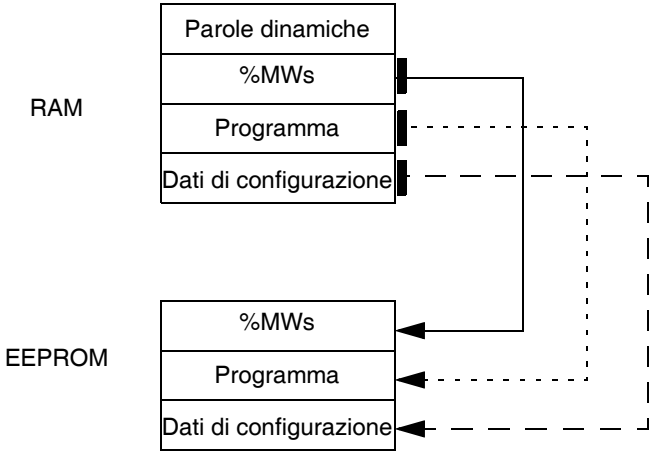
Le informazioni che seguono descrivono in modo dettagliato le funzioni di backup e di ripristino della memoria nei controller modulari e compatti senza cartuccia di .backup e senza inserimento della memoria estesa.

In breve

È possibile eseguire il backup di programmi, parole di memoria e dati di configurazione Twido tramite la EEPROM interna dei controller. Poiché il salvataggio di un programma nella EEPROM interna cancella le parole di memoria salvate in precedenza, è necessario eseguire prima di backup del programma e quindi delle parole di memoria configurate. I dati dinamici possono essere memorizzati in parole di memoria, per poi eseguire il backup nella EEPROM. Se nella EEPROM interna non è salvato alcun programma, non è possibile eseguire il salvataggio delle parole di memoria.

Struttura della memoria

Di seguito è riportato un diagramma con la struttura della memoria del controller. Le frecce mostrano ciò di cui è possibile eseguire il backup nella memoria EEPROM dalla RAM:



Programma Backup

Di seguito è riportata la procedura per eseguire il backup del programma nella EEPROM:

Passo	Azione
1	Devono essere vere le seguenti condizioni: Programma valido nella RAM.
2	Dalla finestra del software Twido, visualizzare il menu alla voce "Controller", scorrere fino a "Backup" e fare clic su di esso.

Ripristino del programma

Durante l'accensione è possibile ripristinare il programma nella RAM dalla EEPROM (purché non siano installate cartucce o memoria estesa):

- Il programma della RAM non è valido.

Per ripristinare manualmente il programma dalla EEPROM procedere come segue:

- Dalla finestra del software Twido, visualizzare il menu alla voce "Controller", scorrere fino a "Ripristina" e fare clic su di esso.

Dati (%MWs) Backup

Di seguito è riportata la procedura per eseguire il backup dei dati (parole di memoria) nella EEPROM:

Passo	Azione
1	Devono essere vere le seguenti condizioni: Nella RAM è presente un programma valido (%SW96:X6=1). Lo stesso programma valido deve essere presente come backup nella EEPROM. Parole di memoria configurate nel programma.
2	Impostare %SW97 in base alla lunghezza delle parole di memoria da salvare. **Nota:** la lunghezza non può superare la lunghezza configurata per le parole di memoria e deve essere compresa tra 0 e 512.
3	Impostare %SW96:X0 a 1.

Ripristino dei dati (%MWs)

Ripristinare %MWs manualmente impostando il bit di sistema %S95 a 1.
Devono essere vere le seguenti condizioni:

- Applicazione di backup valida nella EEPROM
- L'applicazione nella RAM corrisponde all'applicazione di backup nella EEPROM
- Le parole di memoria di backup sono valide.

Backup e ripristino con una cartuccia di backup da 32 K

Introduzione

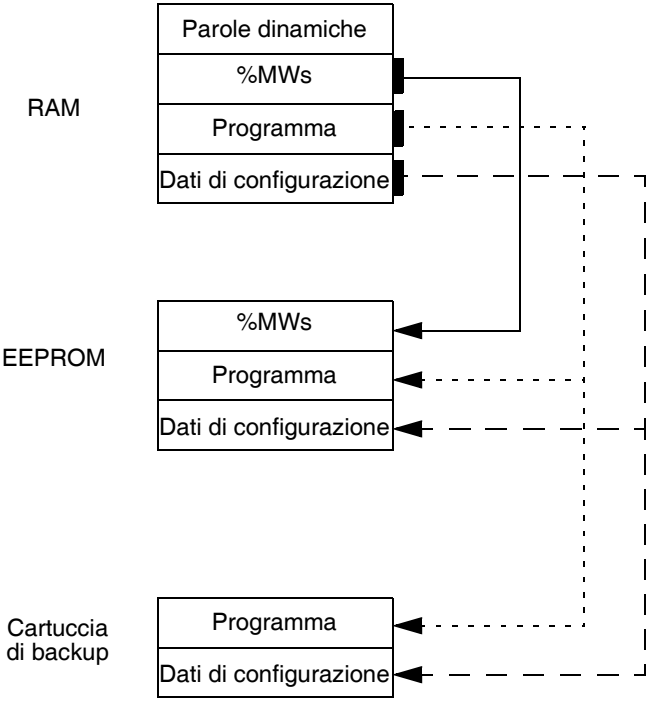
Le informazioni che seguono descrivono in modo dettagliato le funzioni di backup e di ripristino della memoria nei controller modulari e compatti tramite una cartuccia di backup da 32 K.

In breve

La cartuccia di backup consente di salvare un programma e di trasferirlo ad altri controller Twido. Una volta installato o salvato il programma, è necessario rimuoverla dal controller e conservarla a parte. Sulla cartuccia è possibile salvare solo i dati di programma e di configurazione (sulla cartuccia di backup da 32K non è possibile salvare %MWs). I dati dinamici possono essere memorizzati in parole di memoria e quindi è possibile eseguire il backup nella memoria EEPROM. Una volta completa l'installazione del programma, eventuali %MWs di cui si è eseguito il backup nella memoria EEPROM interna prima dell'installazione saranno persi.

Struttura della memoria

Di seguito è riportato un diagramma con la struttura della memoria del controller con collegata la cartuccia di backup. Le frecce mostrano ciò di cui è possibile eseguire il backup nella memoria EEPROM e nella cartuccia dalla RAM:



Programma Backup

Di seguito è riportata la procedura per eseguire il backup del programma nella cartuccia di backup:

Passo	Azione
1	Spegnere il controller.
2	Inserire la cartuccia di backup.
3	Accendere il controller.
4	Dalla finestra del software Twido, visualizzare il menu alla voce "Controller", scorrere fino a "Backup" e fare clic su di esso.
5	Spegnere il controller.
6	Rimuovere la cartuccia di backup dal controller.

Ripristino del programma

Per caricare un programma salvato su una cartuccia di backup in un controller, procedere come segue:

Passo	Azione
1	Spegnere il controller.
2	Inserire la cartuccia di backup.
3	Accendere il controller. Se è configurato l'avvio automatico, è necessario spegnere e riaccendere nuovamente per entrare in modalità Run.
4	Spegnere il controller.
5	Rimuovere la cartuccia di backup dal controller.

Dati (%MWs) Backup

Di seguito è riportata la procedura per eseguire il backup dei dati (parole di memoria) nella EEPROM:

Passo	Azione
1	Devono essere vere le seguenti condizioni: Programma valido nella RAM. Lo stesso programma valido deve essere presente come backup nella EEPROM. Parole di memoria configurate nel programma.
2	Impostare %SW97 in base alla lunghezza delle parole di memoria da salvare. **Nota** la lunghezza non può superare la lunghezza configurata per le parole di memoria e deve essere compresa tra 0 e 512.
3	Impostare %SW96:X0 a 1.

Ripristino dei dati (%MWs)

Ripristinare %MWs manualmente impostando il bit di sistema %S95 a 1.
Devono essere vere le seguenti condizioni:

- Applicazione di backup valida nella EEPROM
- L'applicazione nella RAM corrisponde all'applicazione di backup nella EEPROM
- Le parole di memoria di backup sono valide.

Uso della cartuccia di memoria estesa da 64K

Introduzione

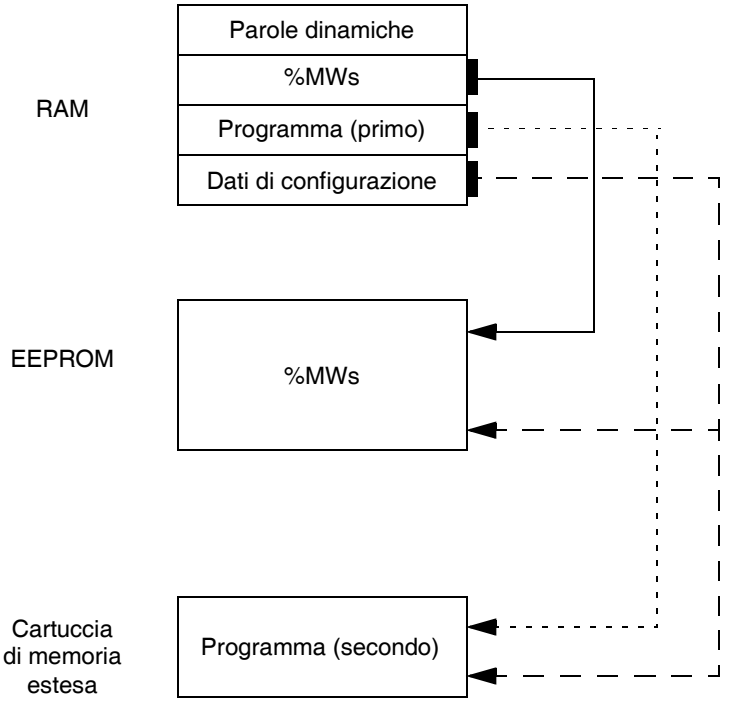
Le informazioni che seguono descrivono in modo dettagliato l'uso delle funzioni di memoria nei controller modulari quando si utilizza una cartuccia di memoria estesa da 64 K.

In breve

La cartuccia di memoria estesa da 64 K consente di estendere la capacità di memoria del programma del controller Twido da 32k a 64K. La cartuccia deve restare inserita nel controller per tutto il tempo in cui si utilizza il programma esteso. Se si rimuove la cartuccia, il controller si blocca. Il backup delle parole di memoria resta nella memoria EEPROM del controller. I dati dinamici possono essere memorizzati in parole di memoria, per poi eseguire il backup nella EEPROM. La cartuccia di memoria estesa da 64K si comporta all'accensione come la cartuccia di backup da 32K.

Struttura della memoria

Di seguito è riportato un diagramma con la struttura della memoria del controller che utilizza una cartuccia di memoria estesa. Le frecce indicano gli elementi salvati nella memoria EEPROM e nella cartuccia di memoria estesa da 64 K dalla RAM:



Configurazione del software e installazione della memoria estesa

Prima di iniziare a scrivere il programma esteso, è necessario installare la cartuccia di memoria estesa da 64K nel controller. Di seguito vengono descritte le quattro operazioni necessarie a questo scopo:

Passo	Azione
1	Nell'opzione di menu Hardware della finestra software Twido, immettere "TWDXCPMFK64".
2	Spegnere il controller.
3	Inserire la cartuccia di memoria estesa da 64K.
4	Accendere il controller.

Salvare il programma.

Una volta installata la cartuccia di memoria estesa da 64K e scritto il programma:

- Dalla finestra del software Twido, visualizzare il menu alla voce "Controller", scorrere fino a "Backup" e fare clic su di esso.

Dati (%MWs) Backup

Di seguito è riportata la procedura per eseguire il backup dei dati (parole di memoria) nella EEPROM:

Passo	Azione
1	Devono essere vere le seguenti condizioni: Presenza di un programma valido. Parole di memoria configurate nel programma.
2	Impostare %SW97 in base alla lunghezza delle parole di memoria da salvare. **Nota:** la lunghezza non può superare la lunghezza configurata per le parole di memoria e deve essere compresa tra 0 e 512.
3	Impostare %SW96:X0 a 1.

Ripristino dei dati (%MWs)

Ripristinare %MWs manualmente impostando il bit di sistema %S95 a 1.
Devono essere vere le seguenti condizioni:

- Presenza di un programma valido.
- Le parole di memoria di backup sono valide.

Modalità operative del controller

4

In breve

Oggetto di questo capitolo

Questo capitolo descrive le modalità operative del controller e l'esecuzione ciclica e periodica del programma. Sono inclusi dettagli sulla mancanza e sul ripristino dell'alimentazione.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Scansione ciclica	66
Scansione periodica	68
Controllo del tempo di scansione	71
Modalità operative	72
Gestione delle interruzioni e del ripristino dell'alimentazione	74
Gestione di un riavvio a caldo	76
Gestione di un avvio a freddo	78
Inizializzazione degli oggetti	80

Scansione ciclica

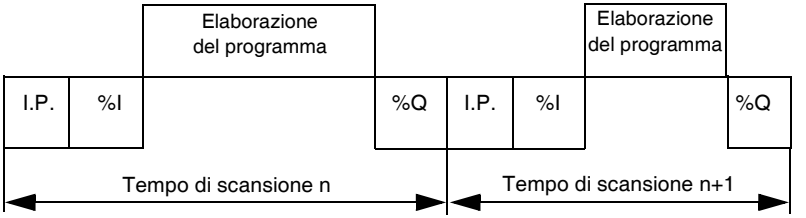
Introduzione

La scansione ciclica consiste nel concatenamento uno dopo l'altro dei cicli del controller. Dopo aver effettuato l'aggiornamento delle uscite (terza fase del ciclo del task), il sistema esegue un certo numero di task propri e attiva immediatamente un altro ciclo di task.

Nota: La durata di scansione del programma utente è controllata dal timer del watchdog del controller e non può superare i 500 ms. In caso contrario, viene visualizzato un errore e il controller passa immediatamente in modalità Halt. Le uscite in questa modalità vengono forzate al loro stato predefinito di sicurezza.

Funzionamento

Il disegno seguente illustra le fasi operative del tempo della scansione ciclica.



Descrizione delle fasi di un ciclo

La tabella che segue descrive le fasi di un ciclo.

Indirizzo	Fase	Descrizione
I.P.	Elaborazione interna	Il sistema controlla implicitamente il controller (gestendo bit e parole di sistema, aggiornando i valori del timer corrente, aggiornando i LED di stato, rilevando gli interruttori di RUN/ STOP, ecc.) ed elabora le richieste da TwidoSoft (modifiche e animazione).
%I, %IW	Acquisizione degli ingressi	Scrittura in memoria dello stato degli ingressi associato ai moduli digitali e analogici.
-	Elaborazione del programma	Esecuzione del programma applicativo scritto dall'utente.
%Q, %QW	Aggiornamento delle uscite.	Scrittura dei bit o delle parole di uscita associati ai moduli digitali e analogici.

Modalità operativa

Controller in modalità RUN, il processore esegue le seguenti operazioni :

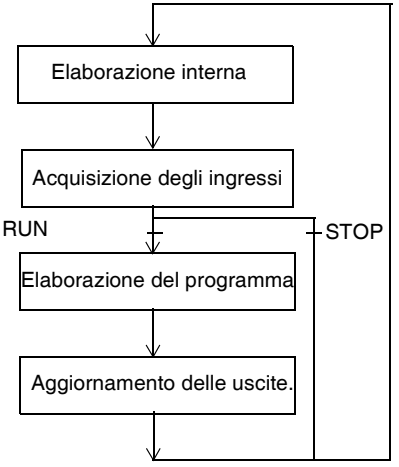
- Elaborazione interna
- Acquisizione degli ingressi
- Elaborazione del programma applicativo
- Aggiornamento delle uscite.

Controller in modalità STOP, il processore esegue le seguenti operazioni :

- Elaborazione interna
- Acquisizione degli ingressi

Illustrazione

L'illustrazione seguente mostra i cicli operativi.



Ciclo di controllo

Il ciclo di controllo viene eseguito dal watchdog.

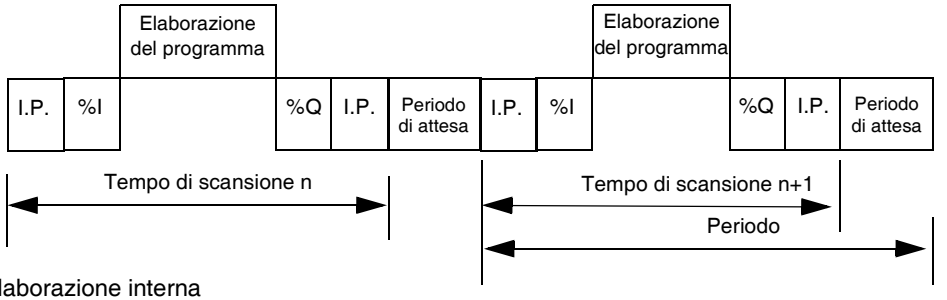
Scansione periodica

Introduzione

In questa modalità operativa, l'acquisizione degli ingressi, l'elaborazione del programma applicativo e l'aggiornamento delle uscite sono operazioni eseguite periodicamente in base al tempo definito al momento della configurazione (a partire da 2-150 ms).
All'inizio della scansione del controller, un timer, il cui valore viene inizializzato al periodo definito nella configurazione, avvia il conteggio alla rovescia. La scansione del controller deve essere conclusa prima della scadenza del timer e prima dell'avvio di una nuova scansione.

Funzionamento

Il disegno seguente illustra le fasi esecutive del tempo della scansione periodica.



T.I: Elaborazione interna

Descrizione delle fasi operative

La tabella che segue descrive le fasi operative.

Indirizzo	Fase	Descrizione
I.P.	Elaborazione interna	Il sistema controlla implicitamente il controller (gestendo bit e parole di sistema, aggiornando i valori del timer corrente, aggiornando i LED di stato, rilevando gli interruttori di RUN/ STOP, ecc.) ed elabora le richieste da TwidoSoft (modifiche e animazione).
%I, %IW	Acquisizione degli ingressi	Scrittura in memoria dello stato degli ingressi associato ai moduli digitali e analogici.
-	Elaborazione del programma	Esecuzione del programma applicativo scritto dall'utente.
%Q, %QW	Aggiornamento delle uscite.	Scrittura dei bit o delle parole di uscita associati ai moduli digitali e analogici.

Modalità operativa

Controller in modalità RUN, il processore esegue le seguenti operazioni:

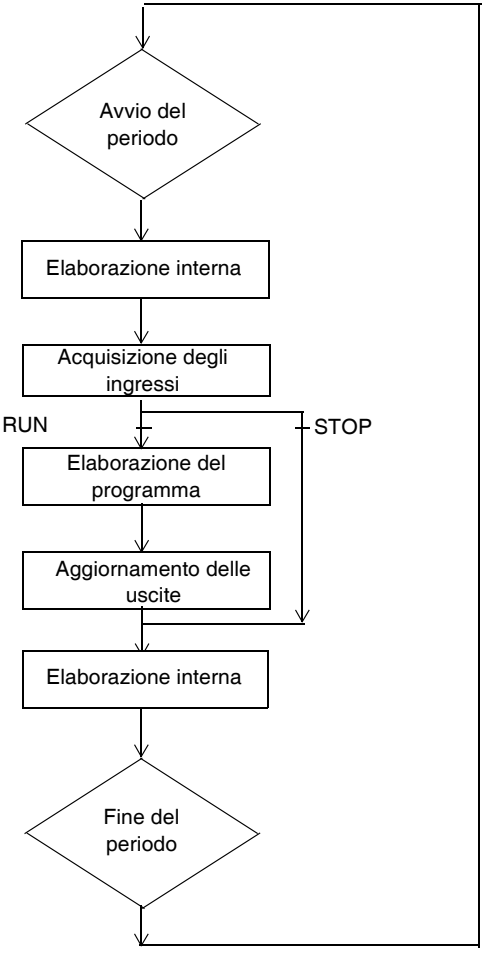
- Elaborazione interna
- Acquisizione degli ingressi
- Elaborazione del programma applicativo
- Aggiornamento delle uscite.

Se il periodo non è concluso, il processore completa il ciclo operativo fino alla fine del periodo di elaborazione interna. Se il tempo operativo è superiore a quello assegnato al periodo, il controller indica che il periodo è stato superato impostando il bit di sistema %S19 a 1. Il processo continua e viene eseguito completamente. Tuttavia, non deve superare il limite di tempo del watchdog. La scansione seguente viene collegata implicitamente dopo la scrittura delle uscite della scansione in corso.

Controller in modalità STOP, il processore esegue le seguenti operazioni :

- Elaborazione interna
- Acquisizione degli ingressi

Illustrazione L'illustrazione seguente mostra i cicli operativi.



Ciclo di controllo Vengono eseguiti due controlli:
- Overflow periodo
- Watchdog

Controllo del tempo di scansione

Punti generali

Il ciclo del task è regolato da un timer watchdog denominato Tmax (durata massima del ciclo del task). Consente la visualizzazione degli errori dell'applicazione (loop infiniti e così via) e garantisce la durata massima per l'aggiornamento delle uscite.

WatchDog software (funzionamento periodico o ciclico)

Nel funzionamento periodico o ciclico, l'attivazione del watchdog provoca un errore software. L'applicazione passa in modalità HALT e il bit di sistema %S11 è messo a 1. Il riavvio del task richiede una connessione a TwidoSoft per l'analisi della causa dell'errore, la modifica dell'applicazione per correggere l'errore e il ripristino del programma in modalità RUN.

Nota: Lo stato HALT si verifica quando l'applicazione viene interrotta immediatamente a causa di un errore software, come ad esempio il superamento della scansione. I dati mantengono i valori correnti, che consentono un'analisi della causa dell'errore. Il programma si arresta sull'istruzione in corso. La comunicazione con il controller è disponibile.

Controllo sul funzionamento periodico

Nel funzionamento periodico viene utilizzato un controllo supplementare per rilevare il periodo superato:

- **%S19** indica che il periodo è stato superato. È impostato a:
  - 1 dal sistema quando il tempo di scansione è superiore al periodo del task,
  - 0 dall'utente.
- **%SW0** contiene il valore del periodo (0-150 ms). Esso è:
  - Inizializzato in caso di avvio a freddo dal valore impostato nella configurazione.
  - Può essere modificato dall'utente.

Utilizzo del tempo di esecuzione del task principale

Le seguenti parole di sistema sono utilizzate per informazioni sul tempo del ciclo di scansione del controller:

- **%SW11** inizializza il tempo di watchdog massimo (da 10 a 500 ms).
- **%SW30** contiene il tempo di esecuzione per l'ultimo ciclo di scansione del controller.
- **%SW31** contiene il tempo di esecuzione per il ciclo di scansione più lungo del controller dall'ultimo avvio a freddo.
- **%SW32** contiene il tempo di esecuzione per il tempo di scansione più corto del controller dall'ultimo avvio a freddo.

Nota: È possibile accedere alle diverse informazioni anche dall'editor della configurazione.

Modalità operative

Introduzione

TwidoSoft viene utilizzato per prendere in considerazione i tre principali gruppi di modalità operative:

- Controllo
- Funzionamento o produzione
- Arresto

Avvio tramite Grafcet

Queste diverse modalità operative sono accessibili tramite Grafcet e applicando i seguenti metodi:

- Inizializzazione Grafcet
- Preimpostazione dei passi
- Mantenimento di una situazione
- Blocco dei grafici

L'elaborazione preliminare e l'uso dei bit di sistema garantiscono la gestione di modalità operative efficaci senza complicare o sovraccaricare il programma utente.

Bit di sistema Grafcet

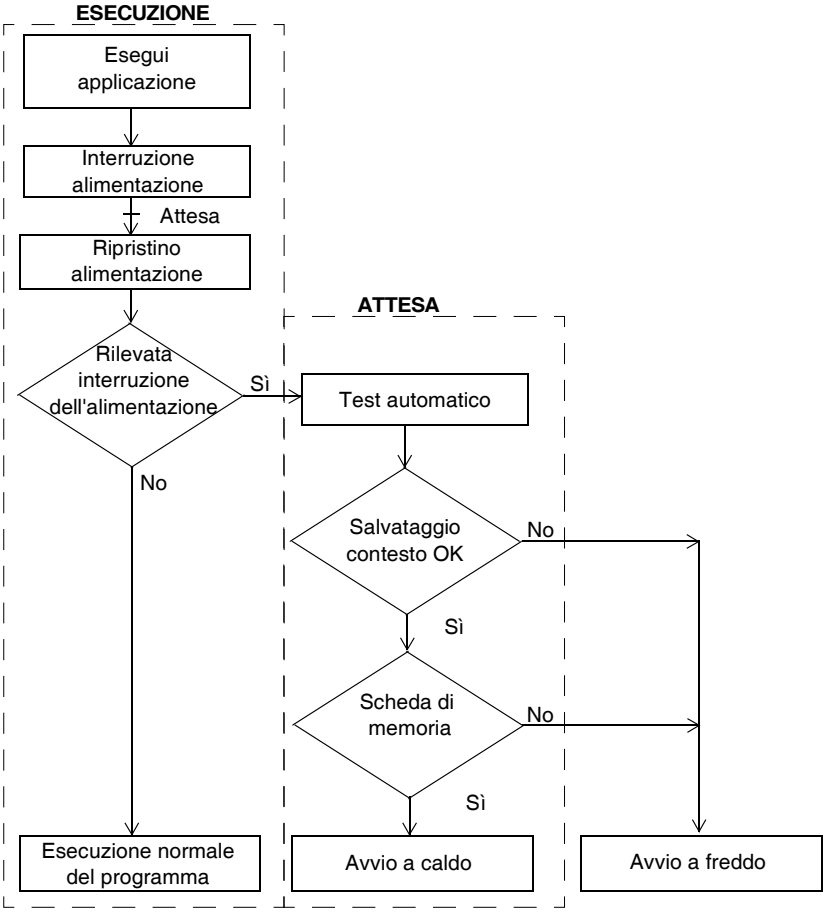
L'uso dei bit %S21, %S22 e %S23 è riservato esclusivamente all'elaborazione preliminare. Questi bit vengono azzerati automaticamente dal sistema e possono essere scritti solo dall'istruzione Set **S**.
La tabella che segue elenca i bit di sistema correlati a Grafcet:

Bit	Funzione	Descrizione
%S21	Inizializzazione GRAFCET	Normalmente impostato a 0, è impostato a 1 tramite: ● avvio a freddo, **%S0=1**; ● l'utente, solo nella parte di programma di preelaborazione, mediante un'istruzione Imposta **S %S21** o Imposta bobina **-(S)- %S21**. Conseguenze: ● Disattivazione di tutti i passi attivi. ● Attivazione di tutti i passi iniziali.
%S22	REIMPOSTA GRAFCET	Normalmente impostato a 0, può essere impostato a 1 solo dal programma durante la preelaborazione. Conseguenze: ● Disattivazione di tutti i passi attivi. ● Interruzione della scansione dell'elaborazione sequenziale.
%S23	Preimpostazione e blocco GRAFCET	Normalmente impostato a 0, può essere impostato a 1 solo dal programma durante la preelaborazione. ● Preposizionamento impostando %S22 a 1. ● Preposizione dei passi da attivare con una serie di istruzioni S Xi. ● Abilita il preposizionamento impostando %S23 a 1. Blocco di una situazione: ● Nella situazione iniziale: mantenendo %S21 a 1 tramite il programma. ● Nella situazione "vuoto": mantenendo %S22 a 1 tramite il programma. ● Nella situazione determinata mantenendo %S23 a 1.

Gestione delle interruzioni e del ripristino dell'alimentazione

Illustrazione

L'illustrazione seguente mostra i vari ripristini dell'alimentazione rilevati dal sistema. Se la durata dell'interruzione è inferiore al tempo del filtro dell'alimentatore (circa 10 ms per l'alimentazione a corrente alternata o 1 ms per l'alimentazione a corrente continua) non viene rilevata dal programma che viene eseguito normalmente.



Nota: il contesto viene salvato in una RAM conservata a batteria. All'accensione, il sistema verifica lo stato della batteria e del contesto salvato per stabilire se può verificarsi un avvio a caldo.

Bit di ingresso Run/Stop e opzione Avvio automatico in Run

Il bit di ingresso Run/Stop ha priorità sull'opzione "Avvio automatico in Run" disponibile dalla finestra di dialogo Modalità di scansione. Se il bit Run/Stop viene impostato a 1, il controller si riavvierà in modalità Run al ripristino dell'alimentazione. La modalità del controller viene determinata nel seguente modo:

Bit di ingresso Run/Stop	Avvio automatico in Run	Stato risultante
Zero	Zero	Stop
Zero	Uno	Stop
Fronte di salita	Indifferente	Run
Uno	Indifferente	Run
Non configurato nel software	Zero	Stop
Non configurato nel software	Uno	Run

Nota: per tutti i tipi di controller Compatti con versione software V1.0, se il controller si trova nella modalità Run quando si interrompe l'alimentazione e il flag di "Avvio automatico in Run" non è stato impostato nella finestra di dialogo di Modalità di scansione, il controller si riavvia nella modalità Stop al ripristino dell'alimentazione, in caso contrario si riavvia a freddo.

Nota: per tutti i tipi di controller Modulari e Compatti con versione software V1.11, se la batteria del controller funziona normalmente quando l'alimentazione viene interrotta, il controller si riavvia nella modalità in atto al momento dell'interruzione dell'alimentazione. Il flag "Avvio automatico in Run", selezionato nella finestra di dialogo Modalità di scansione, non modifica la modalità al ripristino dell'alimentazione.

Funzionamento

La tabella seguente descrive le fasi di elaborazione per le interruzioni di alimentazione.

Fase	Descrizione
1	Nel caso di un'interruzione dell'alimentazione, il sistema memorizza il contesto dell'applicazione e l'ora dell'interruzione.
2	Imposta tutti gli stati di uscita in uno stato di sicurezza (stato 0).
3	Al ripristino dell'alimentazione, il contesto salvato viene confrontato con quello in corso. Tale confronto consente di definire il tipo di riavvio da eseguire: ● se il contesto dell'applicazione è cambiato (perdita del contesto di sistema o nuova applicazione), il controller inizializza l'applicazione nel seguente modo: avvio a freddo (sistematico per il tipo Compatto). ● se il contesto dell'applicazione è lo stesso, il controller viene riavviato senza inizializzazione dei dati: riavvio a caldo.

Gestione di un riavvio a caldo

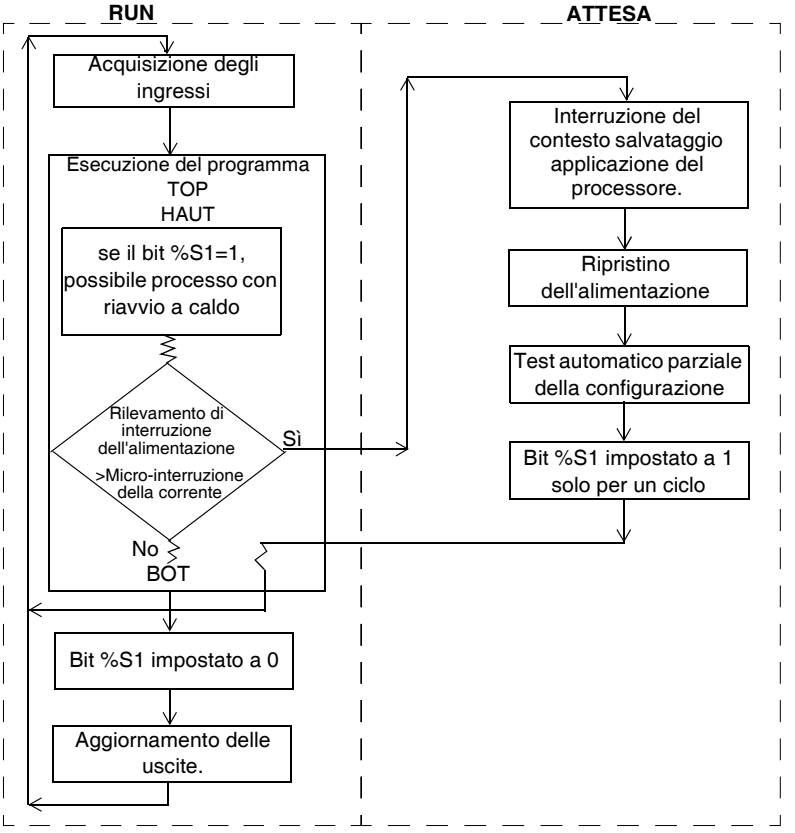
Causa di un riavvio a caldo

Il riavvio a caldo può essere provocato da:

- quando l'alimentazione viene ripristinata senza perdita del contesto di un'applicazione,
- quando il bit **%S1** è impostato a 1 dal programma,
- dal pannello visualizzatore quando il controller è in modalità STOP.

Illustrazione

L'illustrazione seguente descrive un'operazione di riavvio a caldo in modalità RUN.



Riavvio dell'esecuzione di un programma

La tabella seguente descrive le fasi di riavvio per l'esecuzione di un programma dopo un riavvio a caldo.

Fase	Descrizione
1	L'esecuzione del programma riprende dallo stesso elemento in cui si trovava prima dell'interruzione dell'alimentazione, senza aggiornare le uscite. **Nota:** Viene riavviato solo lo stesso elemento dal codice utente. Il codice del sistema (ad esempio, l'aggiornamento delle uscite) non viene riavviato.
2	Alla fine del ciclo di riavvio, il sistema: ● Non riserva l'applicazione, se questa era riservata (e provoca uno STOP in caso di debugging) ● Reinizializza i messaggi
3	Il sistema esegue un ciclo di riavvio in cui: ● riavvia il task con i bit **%S1** (indicatore di riavvio a caldo) e **%S13** (primo ciclo in modalità RUN) impostato a 1, ● azzera i bit **%S1** e **%S13** alla fine del primo ciclo del task.

Elaborazione di un avvio a caldo

Nel caso di un avvio a caldo, se è richiesto un particolare processo applicativo, il bit **%S1** deve essere testato all'inizio del ciclo del task e deve essere richiamato il programma corrispondente.

Uscite dopo una mancanza di alimentazione

Non appena viene rilevata una mancanza di corrente, le uscite sono impostate in uno stato di sicurezza 0 (default).
Quando l'alimentazione viene ripristinata, le uscite si trovano nell'ultimo stato finché non vengono aggiornate nuovamente dal task.

Gestione di un avvio a freddo

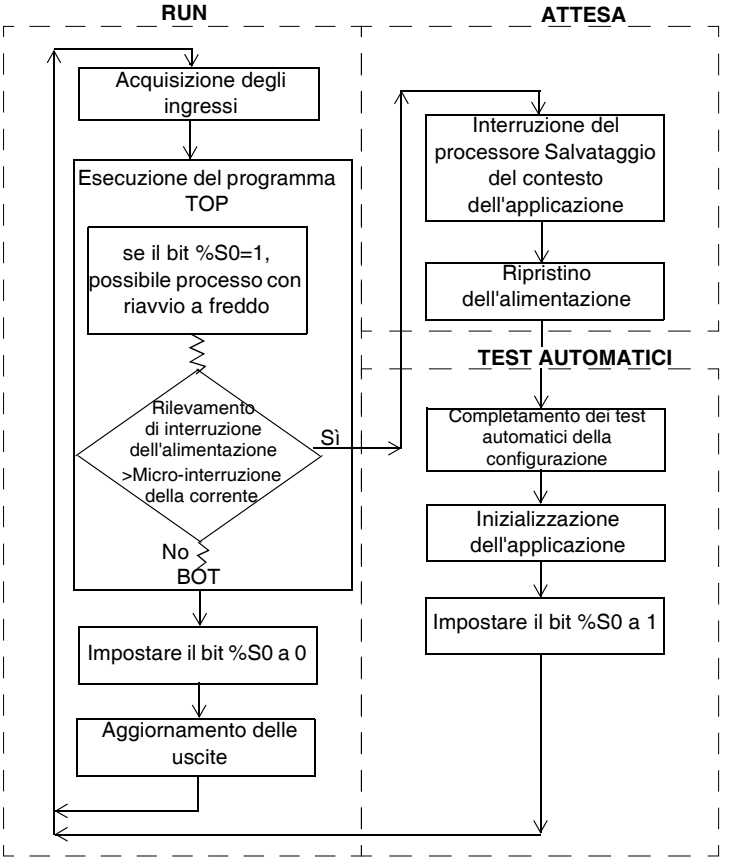
Causa di un avvio a freddo

Un avvio a freddo può verificarsi:

- Durante il caricamento di una nuova applicazione nella RAM.
- Quando l'alimentazione viene ripristinata con perdita del contesto di un'applicazione.
- Quando il bit di sistema **%S0** è impostato a 1 dal programma.
- Dal pannello visualizzatore quando il controller è in modalità STOP.

Illustrazione

L'illustrazione seguente descrive un'operazione di riavvio a freddo in modalità RUN.



Funzionamento

La tabella seguente descrive le fasi di riavvio per l'esecuzione di un programma dopo un riavvio a freddo.

Fase	Descrizione
1	All'avvio, il controller è in modalità RUN. In caso di riavvio successivo a un arresto provocato da un errore, il sistema richiede il riavvio a freddo. L'esecuzione del programma viene riavviata all'inizio del ciclo.
2	Il sistema: ● Reimposta i bit interni, le parole e le immagini di I/O a 0. ● Inizializza i bit e le parole di sistema. ● Inizializza i blocchi funzione dai dati di configurazione.
3	Per questo primo ciclo di riavvio, il sistema: ● Riavvia il task con i bit **%S0** (indicatore di avvio a freddo) e **%S13** (primo ciclo in RUN) impostato a 1. ● Azzera i bit **%S0** e **%S13** alla fine del primo ciclo del task. ● Imposta i bit **%S31** e **%S38** (indicatori di controllo degli eventi) allo stato iniziale 1. ● Azzera il bit **%S39** (indicatore di controllo degli eventi) e la parola **%SW48** (conteggio di tutti gli eventi eseguiti eccetto gli eventi periodici).

Elaborazione di un avvio a freddo

Nel caso di un avvio a freddo, se è richiesto un particolare processo dell'applicazione, il bit **%S0** (a 1) deve essere provato durante il primo ciclo del task.

Uscite dopo una mancanza di alimentazione

Non appena viene rilevata una mancanza di corrente, le uscite sono impostate in uno stato di sicurezza 0 (default).
Alla ripresa dell'alimentazione, le uscite si trovano a zero fino a quando vengono aggiornate con il task.

Inizializzazione degli oggetti

Introduzione

I controller possono essere inizializzati da Twido Soft impostando a 1 i bit di sistema **%S0** (avvio a freddo) e **%S1** (riavvio a caldo).

Inizializzazione con avvio a freddo

Per una inizializzazione con avvio a freddo, il bit di sistema **%S0** deve essere impostato a 1.

Inizializzazione degli oggetti (uguale all'avvio a freddo) alla messa sotto tensione tramite %S0 e %S1

Per un'inizializzazione degli oggetti alla messa sotto tensione, i bit di sistema **%S1** e **%S0** devono essere impostati a 1.

L'esempio seguente mostra come programmare una inizializzazione degli oggetti con riavvio a caldo utilizzando i bit di sistema.




```
LD  %S1   Se %S1 = 1 (riavvio a caldo), impostare %S0 a 1 per inizializzare
          il controller.
ST  %S0   Questi due bit vengono azzerati dal sistema alla fine della
          scansione seguente.
ST  %S9   Questo bit è utilizzato per inizializzare le uscite
```

Nota: Non impostare %S0 a 1 per più di una scansione del controller.

Gestione dei task di evento

5

In breve

In breve

Questo capitolo descrive i task di evento e la loro esecuzione nel controller.

Nota: i task di evento non sono gestiti dal controller Twido TWDLCAA10DRF.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Presentazione dei task di evento	82
Descrizione delle diverse sorgenti di evento	83
Gestione degli eventi	84

Presentazione dei task di evento

Introduzione

Il capitolo precedente presenta i task periodici (Vedi *Scansione periodica, p. 68*) e ciclici (Vedi *Scansione ciclica, p. 66*) in cui gli oggetti vengono aggiornati all'inizio e alla fine del task. Le sorgenti di evento possono provocare interruzioni del task, durante le quali vengono eseguiti task più prioritari (di evento), per permettere un aggiornamento più rapido degli oggetti.
Un task di evento:

- è una parte di programma eseguita al verificarsi di una determinata condizione (sorgente di evento),
- ha priorità più elevata rispetto al programma principale,
- garantisce un tempo di risposta rapida che permette di ridurre il tempo di risposta del sistema.

Descrizione di un evento

Un evento è costituito da:

- una sorgente di evento che può essere definita come una condizione di interruzione software o hardware che interrompe il programma principale (Vedi *Descrizione delle diverse sorgenti di evento, p. 83*),
- una sezione che rappresenta un'entità autonoma di programmazione collegata a un evento,
- una coda di eventi che permette di memorizzare l'elenco degli eventi fino alla loro esecuzione ,
- una priorità che stabilisce l'ordine di esecuzione dell'evento.

Descrizione delle diverse sorgenti di evento

Presentazione delle diverse sorgenti di evento

Una sorgente di evento può essere gestita dal software per accertarsi che il programma principale venga interrotto correttamente dall'evento e per richiamare la sezione di programmazione collegata all'evento. Il tempo di scansione dell'applicazione non ha alcun effetto sull'esecuzione degli eventi.
Sono ammesse le seguenti nove sorgenti di evento:

- 4 condizioni collegate alle soglie dei blocchi funzione contatore molto veloce (2 eventi per istanza di %VFC),
- 4 condizioni collegate agli ingressi fisici di un controller di base,
- 1 condizione periodica.

Una sorgente di evento può essere associata a un solo evento e deve essere rilevata immediatamente dal software TwidoSoft. Non appena rilevata, il software esegue la sezione di programmazione associata all'evento: ogni evento è associato a una subroutine con l'etichetta **SRi:** definita durante la configurazione delle sorgenti di evento.

Eventi sugli ingressi fisici di un controller di base

È possibile utilizzare gli ingressi %I0.2, %I0.3, %I0.4 e %I0.5 come sorgenti di evento, purché non siano bloccati e purché gli eventi siano stati ammessi durante la configurazione.
Le elaborazioni degli eventi possono essere attivate dagli ingressi da 2 a 5 di un controller di base (posizione 0), su fronte di salita o di discesa.
Per ulteriori informazioni sulla configurazione dell'evento, consultare la sezione "Configurazione hardware -> Configurazione degli ingressi" della guida in linea "Guida all'uso di TwidoSoft".

Eventi sulle uscite di un blocco funzione %VFC

Le uscite TH0 e TH1 del blocco funzione %VFC sono sorgenti di evento. Le uscite TH0 e TH1 passano rispettivamente:

- a 1 quando il valore è superiore alla soglia S0 e alla soglia S1,
- a 0 quando il valore è inferiore alla soglia S0 e alla soglia S1.

Un fronte di salita o di discesa di queste uscite può attivare un'elaborazione degli eventi.
Per ulteriori informazioni sulla configurazione dell'evento, consultare la sezione "Configurazione software-> Contatori molto veloci" della guida in linea "Guida all'uso di TwidoSoft".

Evento periodico

Questo evento esegue periodicamente una stessa sezione di programmazione. Questo task è prioritario rispetto al task principale (master).
Questa sorgente di evento, al contrario, non è prioritaria rispetto alle altre sorgenti di evento.
Il periodo di questo task è fissato al momento della configurazione ed è compreso tra 5 e 255 ms. Può essere usato un solo evento periodico.
Per ulteriori informazioni sulla configurazione di questo evento, consultare la sezione "Configurazione dei parametri di programma -> Modalità di scansione" della guida in linea "Guida all'uso di TwidoSoft".

Gestione degli eventi

Code di eventi e priorità

Gli eventi possono avere due priorità: alta o bassa. Un **solo** tipo di evento, e quindi una sola sorgente di evento, tuttavia, può avere priorità alta. Gli altri eventi devono pertanto avere priorità bassa e il loro ordine di esecuzione dipende dal rispettivo ordine di rilevamento.
Per gestire l'ordine di esecuzione dei task di evento, esistono due code di eventi:
- una consente di memorizzare fino a 16 eventi di priorità alta di una stessa sorgente di evento,
- l'altra permette di memorizzare fino a 16 eventi di priorità bassa delle altre sorgenti di evento,

Queste code sono gestite secondo il criterio FIFO (First In First Out): il primo evento memorizzato viene eseguito per primo. Poiché è possibile memorizzare solo 16 eventi, gli eventi supplementari vengono perduti.
La coda di priorità bassa viene eseguita solo dopo che la fila di priorità alta è vuota.

Gestione delle code di eventi

Ogni volta che si verifica un'interruzione, collegata a una sorgente di evento, viene avviata questa sequenza:

Passo	Descrizione
1	Gestione dell'interruzione: ● rilevamento dell'interruzione fisica, ● evento memorizzato nella coda di eventi appropriata, ● verifica che non sia già in corso un evento avente la medesima priorità, nel qual caso l'evento resta in attesa nella propria coda.
2	Salvataggio del contesto.
3	Esecuzione della sezione di programmazione (sottoprogramma denominato SRi:) collegata all'evento.
4	Aggiornamento delle uscite.
5	Ripristino del contesto.

Prima che il contesto venga ripristinato, è necessario che tutti gli eventi della coda vengano eseguiti.

Controllo degli eventi

Per controllare gli eventi (Vedi *Bit di sistema e parole di sistema, p. 603*) sono utilizzati bit e parole di sistema:
- %S31: permette di eseguire o ritardare un evento,
- %S38: permette di inserire o togliere un evento dalla coda degli eventi,
- %S39: permette di sapere se vi sono eventi perduti,
- %SW48: contiene il numero di eventi eseguiti dopo l'ultimo riavvio a freddo (conteggio di tutti gli eventi esclusi quelli periodici).

Il valore del bit %S39 e della parola %SW48 è inizializzato a zero e quello di %S31 e %S38 è impostato allo stato iniziale 1 in occasione di un riavvio a freddo o dopo il caricamento di un'applicazione ma resta invariato in caso di riavvio a caldo. In ogni caso, la coda di eventi viene inizializzata.

Funzioni speciali

II

In breve

Contenuto della sezione

Questa sezione descrive i diversi tipi di comunicazione, le funzioni analogiche integrate, la gestione dei moduli di I/O analogici, l'installazione del bus AS-Interface V2 e il bus di campo CANopen per i controller Twido.

Contenuto di questa parte

Questa parte contiene i seguenti capitoli:

Capitolo	Titolo del capitolo	Pagina
6	Comunicazioni	87
7	Incorporate, funzioni analogiche	189
8	Gestione dei moduli analogici	193
9	Implementazione del bus AS-Interface V2	205
10	Installazione e configurazione del bus di campo CANopen	239
11	Configurazione del gateway Ethernet TwidoPort	281
12	Funzionamento del visualizzatore	309

Comunicazioni

6

In breve

Oggetto di questo capitolo

Questo capitolo fornisce una panoramica sulla configurazione, la programmazione e la gestione delle comunicazioni disponibili con i controller Twido.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Presentazione dei diversi tipi di comunicazione	88
Comunicazioni tra TwidoSoft e il controller	90
Comunicazioni tra TwidoSoft e un modem	96
Comunicazioni in collegamento remoto	108
Comunicazioni ASCII	120
Comunicazioni Modbus	132
Richieste standard Modbus	150
Classe di implementazione Transparent Ready (Twido Serial A05, Ethernet A15)	155
Panoramica sulla comunicazione Ethernet TCP/IP	156
Guida alla configurazione TCP/IP rapida per la comunicazione Ethernet tra PC e controller	158
Connessione del controller alla rete	164
Indirizzamento IP	165
Assegnazione di indirizzi IP	166
Configurazione TCP/IP	170
Scheda Configura indirizzo IP	172
Scheda IP contrassegnato	175
Scheda Timeout	177
Scheda Dispositivi remoti	179
Visualizzazione della configurazione Ethernet	181
Gestione delle connessioni Ethernet	182
Indicatori LED Ethernet	184
Messaggeria TCP Modbus	185

Presentazione dei diversi tipi di comunicazione

In breve

Twido fornisce una o due porte di comunicazione seriale per i controller remoti, i controller peer le periferiche generiche. Una delle due porte, se disponibile, può essere utilizzata per qualsiasi servizio, ad eccezione della comunicazione con TwidoSoft, che può essere eseguita solo tramite la prima porta. Ciascun controller Twido supporta tre diversi protocolli di base: collegamento remoto, ASCII o Modbus (Modbus master o Modbus slave).
Inoltre il controller compatto TWDLCAE40DRF presenta una porta di comunicazione Ethernet RJ-45. Supporta il protocollo client/server TCP/IP per la comunicazione peer-to-peer tra controller tramite la rete Ethernet.

Collegamento remoto

Il collegamento remoto è un bus master/slave molto rapido, studiato per trasmettere una piccola quantità di dati tra il controller master e un massimo di sette controller remoti (slave). I dati dell'applicazione o i dati di I/O vengono trasferiti in base alla configurazione dei controller remoti. È possibile combinare vari tipi di controller remoti. Alcuni possono essere controller di I/O remoti, altri controller di estensione.

ASCII

Il protocollo ASCII è un protocollo half-duplex in modalità carattere semplice utilizzato per trasmettere e/o ricevere una stringa di caratteri da o verso una periferica, ad esempio stampante o terminale. Questo protocollo è supportato solo tramite l'istruzione "EXCH".

Modbus

Il protocollo Modbus è un protocollo master/slave che consente a un solo master di richiedere risposte agli slave o di agire in base alla richiesta. Il master può rivolgersi a un singolo slave o può trasmettere un messaggio in diffusione broadcast a tutti gli slave. Gli slave rispondono con un messaggio (risposta) alle richieste che vengono inviate loro singolarmente. Non vengono rinviate le risposte alle query provenienti da un master.
Modbus master - La modalità Modbus master permette al controller Twido di inviare a uno slave una richiesta Modbus e di attendere la risposta da parte dello slave. La modalità Modbus master è supportata solo tramite l'istruzione "EXCH". La modalità Modbus master supporta sia Modbus ASCII sia Modbus RTU.
Modbus slave - La modalità Modbus slave consente al controller Twido di rispondere alle query modbus da un modbus master ed è la modalità di comunicazione di default se non viene configurato un altro tipo di comunicazione. Il controller Twido supporta i dati Modbus e le funzioni di controllo e le estensioni dei servizi per l'accesso agli oggetti. La modalità Modbus slave supporta sia Modbus ASCII sia Modbus RTU.

Nota: su una rete RS-485 è possibile installare 32 dispositivi (senza ripetitori, 1 master e fino a 31 slave), i cui indirizzi possono essere compresi tra 1 e 247.

Modbus TCP/IP

Nota: Modbus TCP/IP è supportato solo dalla serie di controller compatti TWDLCAE40DRF con interfaccia di rete Ethernet integrata.

Di seguito viene descritto il protocollo MBAP (Modbus Application Protocol).
Il protocollo MBAP (Modbus Application Protocol) è un protocollo di livello 7 che fornisce comunicazione peer-to-peer tra controller logici (PLC) e altri nodi di una LAN. L'implementazione corrente del controller Twido TWDLCAE40DRF trasporta il protocollo MBAP tramite TCP/IP sulla rete Ethernet. Le transazioni del protocollo Modbus sono tipiche coppie di messaggi richiesta-risposta. Un controller può essere sia client che server a seconda se sta inviando una richiesta o rispondendo.

Comunicazioni tra TwidoSoft e il controller

In breve

Ciascun controller Twido dispone sulla porta 1 di una presa terminale EIA RS-485 incorporata con alimentatore interno. La porta 1 deve essere utilizzata per la comunicazione con il software di programmazione TwidoSoft.
Per questa connessione non è possibile utilizzare alcuna cartuccia o modulo di comunicazione opzionale. Non è possibile utilizzare nemmeno i modem.
Esistono vari modi per collegare il PC alla porta 1 RS-485 del controller Twido.

- tramite cavo TSXPCX;
- tramite linea telefonica: connessione MODEM.

Inoltre il controller compatto TWDLCAE40DRF dispone di una porta di connessione di rete Ethernet RJ-45 integrata che può essere usata per comunicare con il PC che supporta Ethernet e che esegue il software di programmazione TwidoSoft.
Il PC che supporta Ethernet può comunicare in due modi con la porta RJ-45 del controller Twido TWDLCAE40DRF:

- Per connessione diretta tramite un cavo incrociato Ethernet UTP Cat5 RJ45 (non consigliato),
- Per connessione alla rete Ethernet tramite un cavo Ethernet SFTP Cat5 RJ45 disponibile dal catalogo Schneider Electric (riferimento: 490NTW000••).

⚠ ATTENZIONE

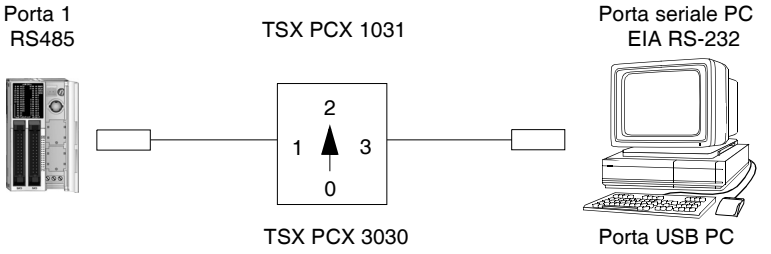
DANNI ALLE APPARECCHIATURE

TwidoSoft potrebbe non rilevare la disconnessione che si verifica spostando fisicamente il cavo di comunicazione TSXPCX1031, o TSX PCX 3030 o Ethernet da un controller a un altro controller. Per evitare questa condizione, usare TwidoSoft per scollegarsi prima di spostare il cavo.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

Collegamento del cavo TSXPCX

La porta EIA RS-232C o USB sul personal computer è collegata alla porta 1 del controller tramite il cavo di comunicazione multifunzione TSXPCX1031 o TSX PCX 3030. Questo cavo, che converte i segnali tra EIA RS-232 e EIA RS-485 per il TSX PCX 1031 e tra USB ed EIA RS-485 per il TSX PCX 3030, è munito di un interruttore a 4 posizioni per selezionare le diverse modalità operative. Le quattro posizioni dell'interruttore sono "0-3" e l'impostazione corretta per la comunicazione tra TwidoSoft e il controller Twido è la posizione 2.
Questa connessione è illustrata nel diagramma seguente.



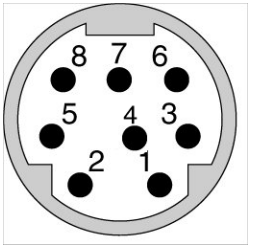
Nota: per questo cavo il segnale DPT sul pin 5 non è a 0V. Ciò indica al controller che si tratta di una connessione TwidoSoft. Il segnale viene regolato internamente per indicare al firmware che si tratta di una connessione TwidoSoft.

Pin di uscita dei connettori maschio e femmina

La figura seguente mostra i pin di uscita di un connettore maschio mini DIN a 8 pin e di una morsettiera.

Mini DIN

TWD NAC232D, TWD NAC485D
TWD NOZ485D, TWD NOZ232D

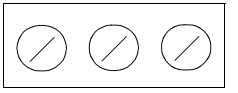


Pin	Base RS485	Opzione RS485	RS232-C
1	D1 (A+)	D1 (A+)	RTS
2	D0 (B-)	D0 (B-)	DTR
3	NC	NC	TXD
4	/DE	NC	RXD
5	/DPT	NC	DSR
6	NC	NC	GND
7	0 V	0 V	GND
8	5 V	5 V	5 V

Nota: consumo totale massimo per modalità a 5V (pin 8): 180mA

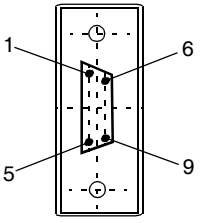
Morsettiera

TWD NAC485T
TWD NOZ485T



Pin	RS485
A	D1 (A+)
B	D0 (B-)
SG	0V

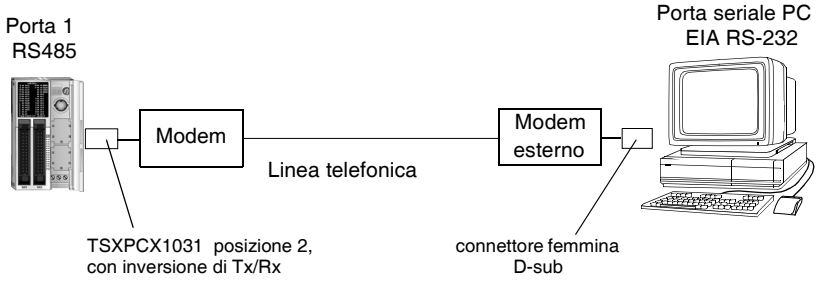
La figura seguente mostra i pin di uscita di un connettore femmina SubD a 9 pin per TSX PCX 1031.



Pin	RS232
1	DCD
2	RX
3	TX
4	DTR
5	SG
6	NC
7	RTS
8	CTS
9	NC

Connessione tramite linea telefonica

La connessione via modem (Vedi *Comunicazioni tra TwidoSoft e un modem, p. 96*) consente la programmazione e la comunicazione con un controller tramite linea telefonica.
Il modem collegato al controller è un dispositivo di **ricezione** collegato alla porta 1 del controller. Il modem collegato al PC può essere interno o esterno, in questo caso connesso tramite porta seriale COM.
Questa connessione è illustrata nel diagramma seguente.



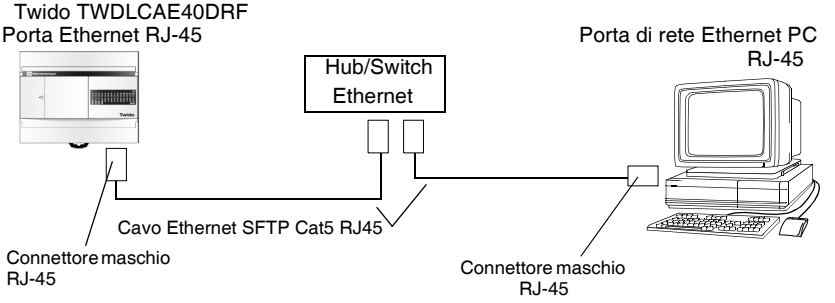
Nota: è possibile collegare un solo modem alla porta 1 del controller.

Nota: Attenzione: non dimenticare di installare il software fornito con il modem, in quanto TwidoSoft tiene conto solo dei modem installati.

Connessione di rete Ethernet

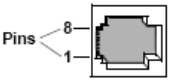
Nota: sebbene la connessione diretta via cavo (tramite un cavo incrociato Ethernet) sia supportata tra il controller Twido TWDLCAE40DRF e il PC che esegue il software di programmazione TwidoSoft, non è un modo di connessione consigliato. Si dovrebbe pertanto preferire una connessione tramite un hub/switch di rete Ethernet.

La figura seguente mostra una connessione PC-Twido tramite un hub/switch di rete Ethernet:



Nota: il PC che esegue l'applicazione TwidoSoft deve supportare Ethernet.

Il controller Twido TWDLCAE40DRF dispone di un connettore RJ-45 per la connessione di rete Ethernet 100 BASE-TX con negoziazione automatica. Supporta velocità di rete di 100M bps e 10 Mbps.
La figura seguente mostra il connettore RJ-45 del controller Twido.



Gli otto pin del connettore RJ-45 sono disposti verticalmente e numerati in ordine dal basso verso l'alto. L'assegnazione dei pin del connettore RJ-45 è riportata nella tabella seguente:

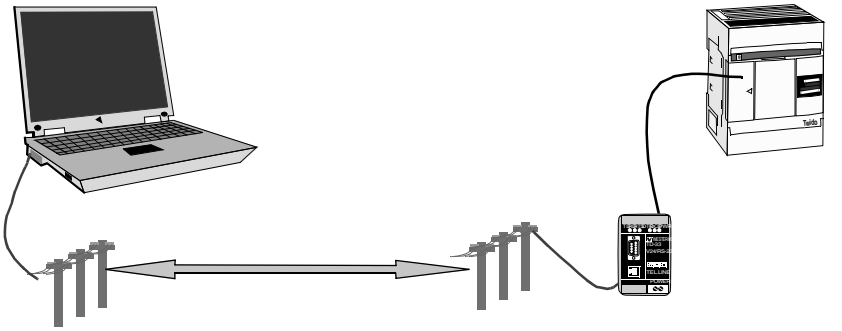
Assegnazione dei pin	Funzione	Polarità
8	NC	
7	NC	
6	RxD	(-)
5	NC	
4	NC	
3	RxD	(+)
2	TxD	(-)
1	TxD	(+)

Nota:
- Lo stesso connettore e la stessa assegnazione dei pin sono usati sia per la rete 10Base-T che per la rete 100Base-TX.
- Per la connessione del controller Twido a una rete 100Base-TX, è necessario usare un cavo Ethernet che sia almeno di categoria 5.

Comunicazioni tra TwidoSoft e un modem

Generale

È possibile collegare un PC che esegue Twidosoft a un controller Twido per trasferire applicazioni, animare oggetti o eseguire comandi in modalità operatore. È possibile inoltre collegare il controller Twido ad altri dispositivi, ad esempio un altro controller Twido, per stabilire la comunicazione con il processo dell'applicazione.

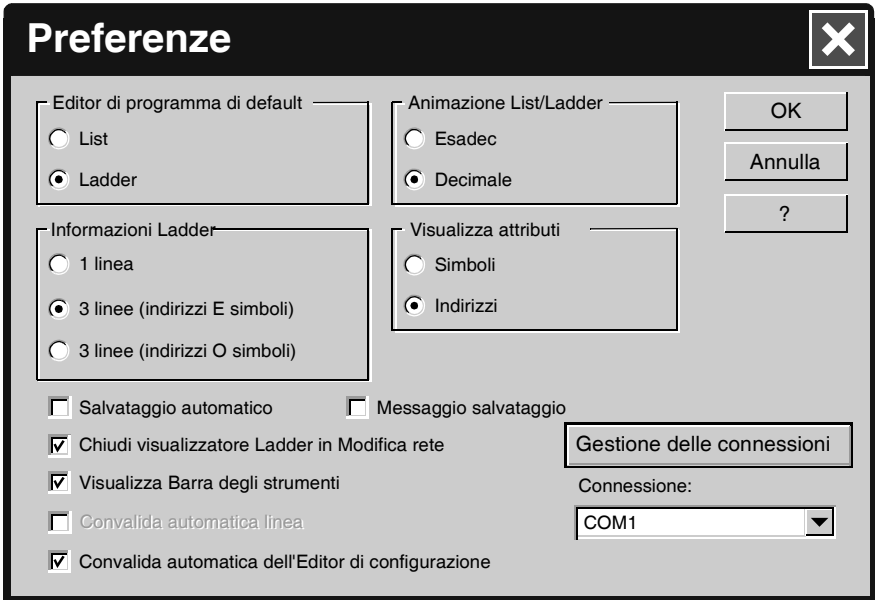


Installazione del modem

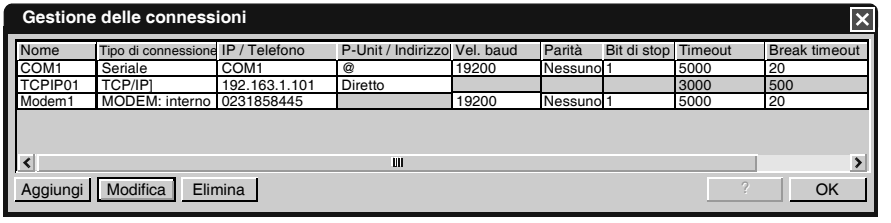
Tutti i modem che l'utente desidera utilizzare con Twidosoft devono essere installati in ambiente Windows dal PC.
Per installare i modem in ambiente Windows, procedere come indicato dalla documentazione di Windows.
Si tratta infatti di un'installazione indipendente da Twidosoft.

Creazione della connessione

La connessione di comunicazione predefinita tra Twidosoft e il controller Twido avviene tramite una porta di comunicazione seriale, utilizzando il cavo TSX PCX 1031 e un adattatore null modem (vedere l'*Appendice 1, p. 105*).
Se un modem viene utilizzato per collegare il PC, è necessario specificarlo nel software Twidosoft.
Per selezionare una connessione con Twidosoft, aprire il menu "File", "Preferenze".



Questa schermata permette di selezionare una connessione o di gestire connessioni, ad esempio per la creazione o la modifica.
Per utilizzare una connessione esistente, selezionarla tra quelle presenti nel menu a discesa.
Per aggiungere, modificare o eliminare una connessione, fare clic una volta su "Gestione delle connessioni". Viene visualizzata una finestra che riporta l'elenco delle connessioni e le relative proprietà.



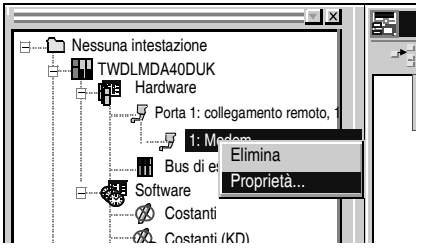
In questo caso, sono elencate due porte seriali (Com1 e Com4) e una connessione modem che utilizza un modello TOSHIBA V.90, configurato per comporre il numero: 0231858445 (chiamata nazionale).
Per poter eseguire la manutenzione dell'applicazione, è possibile modificare il nome di ogni connessione, ad esclusione delle porte COM1 e COM4, il cui nome non è modificabile.
Di seguito è descritto il modo per definire e selezionare la connessione che si desidera utilizzare per collegare il PC a un modem.
Questa procedura, tuttavia, non esaurisce le operazioni necessarie per stabilire una connessione globale tra il computer e il controller Twido.
La tappa successiva riguarda il controller Twido. Il controller Twido remoto deve essere collegato a un modem.
Per poter stabilire una connessione, è necessario che tutti i modem vengano inizializzati. Il controller Twido che utilizza un firmware di versione minima V2.0 è in grado di inviare all'accensione una stringa adattata al modem, purché quest'ultimo sia stato configurato nell'applicazione.

Configurazione del modem

Per configurare un modem in un controller Twido, procedere come segue:



Dopo aver configurato il modem sulla porta 1, è necessario definirne le proprietà. Fare clic con il pulsante destro del mouse sul modem per visualizzare il menu contestuale che propone le azioni "Elimina" o "Proprietà". L'opzione Proprietà permette di selezionare un modem noto, crearne uno nuovo o modificarne uno esistente.



Nota: per la gestione del modem, il controller Twido utilizza esclusivamente la porta 1. Ciò significa che è possibile collegare un modem sulla porta 2 di comunicazione ma, a differenza di quanto avviene se si utilizza la porta 1, in questo caso è necessario eseguire manualmente tutte le modalità di funzionamento e la sequenza di inizializzazione del modem.

Successivamente, selezionare "Proprietà":



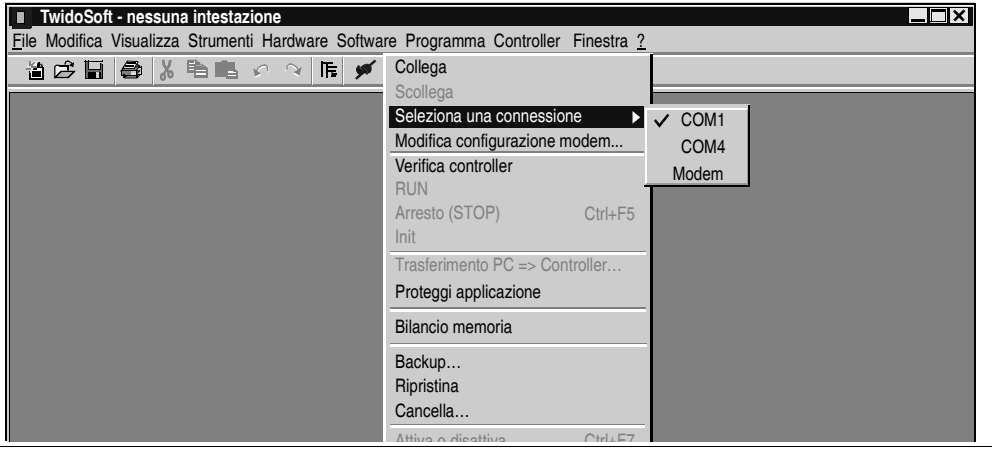
A questo punto, è possibile selezionare un modem definito in precedenza oppure crearne uno nuovo facendo clic sul pulsante "...".



Attribuire un nome al nuovo profilo e completare il comando Hayes d'inizializzazione come descritto nella documentazione del modem.
In questa immagine, "xxxxxx" rappresenta la sequenza di inizializzazione da immettere per preparare il modem per la comunicazione, ovvero velocità in Baud, parità, bit di stop e modalità di ricezione.
Per completare questa sequenza, consultare la documentazione del modem.
La lunghezza massima della stringa è di 127 caratteri.
Una volta terminata l'applicazione o almeno dopo aver completato la descrizione della porta 1 di comunicazione, trasferire l'applicazione utilizzando una "connessione punto-punto".
A questo punto, il controller Twido è pronto per essere collegato tramite i modem a un PC che esegue Twidosoft.

Sequenza di connessione

Una volta preparato Twidosoft e il controller Twido, stabilire la connessione procedendo come segue:

Passo	Azione
1	Accendere il controller Twido e il modem.
2	Avviare il computer e lanciare Twidosoft.
3	Fare clic sul menu "Controller" e sull'opzione "Selezionare una connessione". Selezionare "Modem utilizzato" (o il nome assegnato alla connessione modem utilizzata; vedere "Creazione di una connessione") 
4	Collegare TwidoSoft.

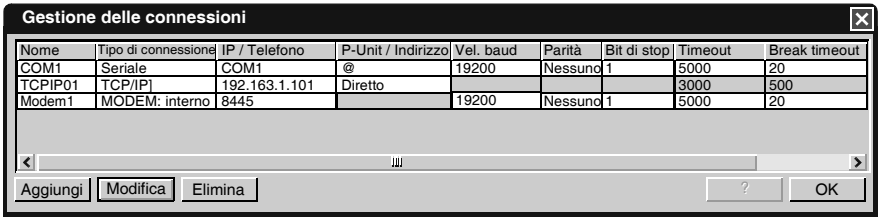
Nota: se si desidera utilizzare sempre la connessione modem, nel menu "File", "Preferenze", selezionare "Modem utilizzato" (o il nome ad esso assegnato). In tal modo Twidosoft memorizza la preferenza.

Modalità operative

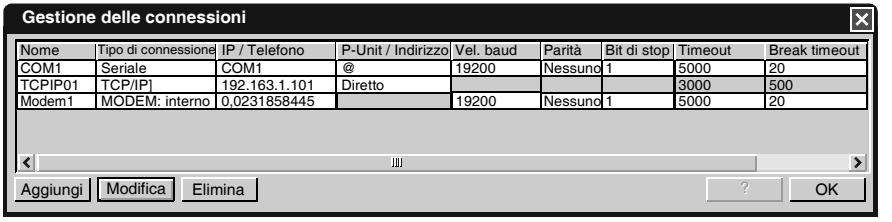
Il controller Twido invia la stringa di inizializzazione al modem collegato sotto tensione. Quando un modem è configurato nell'applicazione Twido, il controller invia prima di tutto un comando "FF" per rilevare se il modem è collegato. Se riceve risposta, il controller invia la stringa di inizializzazione.

Chiamate interne, esterne e internazionali

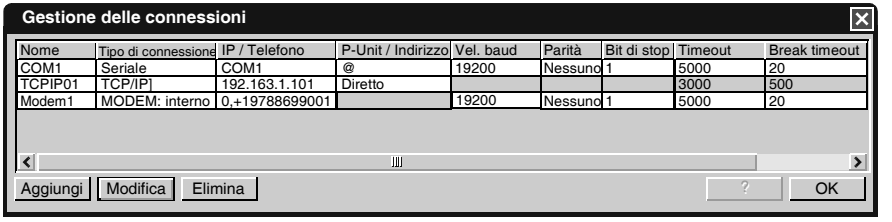
Se la comunicazione con il controller Twido avviene all'interno dell'azienda, è possibile utilizzare solo il numero dell'interno da chiamare, ad esempio: 8445



Se per chiamare i numeri esterni all'azienda si utilizza un centralino che richiede l'uso di "0" o di "9" prima del numero di telefono, utilizzare la seguente sintassi: 0,0231858445 oppure 9,0231858445

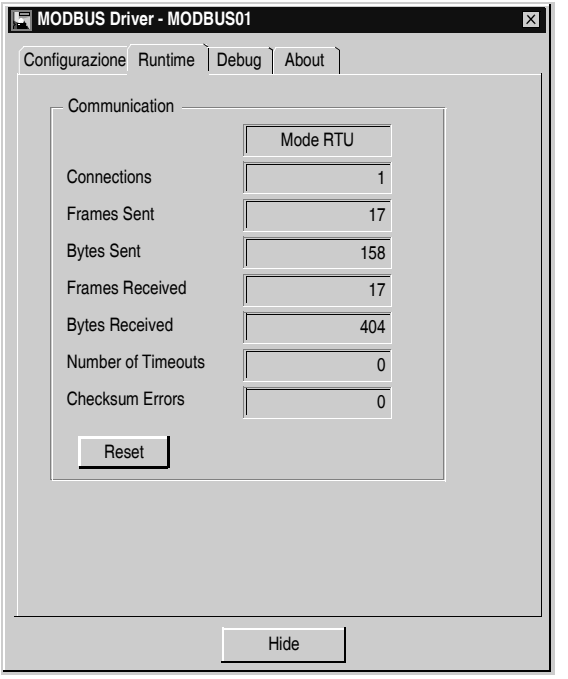


Per le chiamate internazionali, la sintassi è la seguente: +19788699001, ad esempio. Se si utilizza un centralino: 0,+ 19788699001



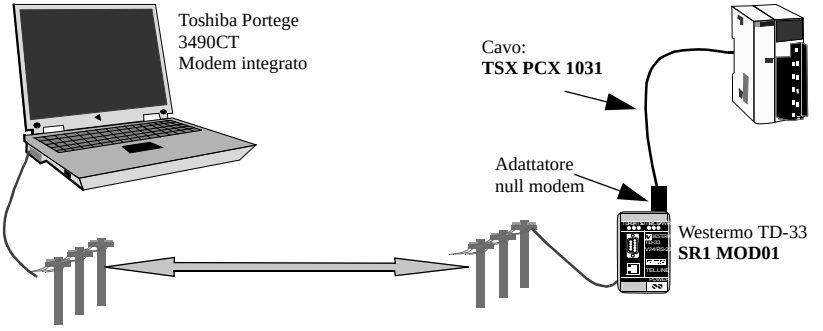
Domande frequenti

Quando la comunicazione è stabilita solo da alcuni minuti, è possibile che si verifichino errori di comunicazione. In questo caso, occorre regolare i parametri di comunicazione.
Per comunicare tramite le porte seriali o i modem interni, Twidosoft utilizza un driver Modbus. Una volta che la comunicazione è iniziata, il driver viene visualizzato nella barra degli strumenti. Fare doppio clic sull'icona del driver per visualizzare la finestra. A questo punto, è possibile accedere ai parametri del driver Modbus. La scheda "Runtime" visualizza le informazioni riguardanti i frame scambiati con il controller remoto.
Se il "Numero di timeout" aumenta o è diverso da 0, modificarne il valore con l'opzione "Gestione delle connessioni", accessibile da Twidosoft selezionando il menu "File", "Gestione delle connessioni". Fare clic sul campo "Timeout" e quindi sul pulsante di modifica e immettere un valore più elevato. Il valore predefinito è "5000", in millesimi di secondi.
Provare nuovamente la connessione. Regolare il valore in modo che la connessione risulti stabile.



Esempi

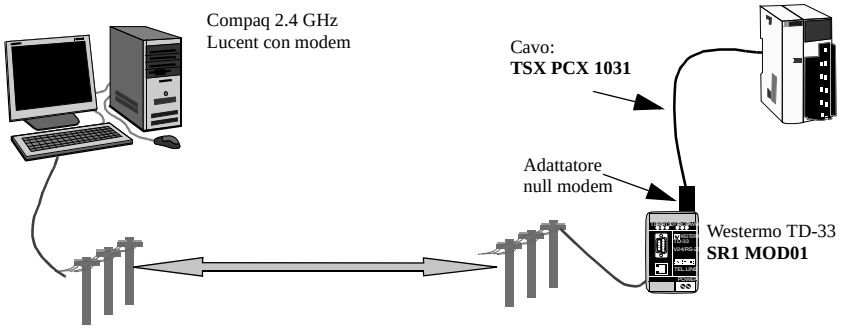
- **Esempio 1**: Twidosoft collegato a TWD LMDA 20DRT (Windows 98 SE).
  - PC: Toshiba Portege 3490CT con Windows 98,
  - Modem (interno al PC): Modem interno Toshiba V.90,
  - Controller Twido: TWD LMDA 20DRT Versione 2.0,
  - Modem (collegato a Twido): tipo Westermo TD-33/V.90 codice di riferimento SR1 MOD01 disponibile nel nuovo catalogo Twido (settembre 03) (vedere l'*Appendice 2, p. 106*),
    Solo per i clienti del Nord America: In quest'area è disponibile il modem TD-33/V.90 US,
  - Cavo: TSX PCX 1031 collegato alla porta 1 di comunicazione Twido e un adattatore: 9 pin maschio/9 pin maschio per incrociare Rx e Tx durante la connessione tra il modem Westermo e il controller Twido (vedere l'*Appendice 1, p. 105*). È possibile anche utilizzare il cavo TSX PCX 1130 (conversione RS485/232 e incrocio Rx/Tx).



Come prima verifica, utilizzare 2 linee telefoniche analogiche interne all'azienda, senza utilizzare il numero di telefono completo ma solo l'interno. È per questo che vi sono solo 4 cifre per il numero di telefono del modem Toshiba V.90 interno. Per questo test, i parametri di connessione (Twidosoft menu "Preferenze", "Gestione delle connessioni") erano impostati sui valori predefiniti, cioè timeout = 5000 e timeout interruzione = 20.

- **Esempio 2**: Twidosoft collegato a TWD LMDA 20DRT (Windows XP Pro)
  - PC: Compaq Pentium 4, 2,4 GHz,
  - Modem: modem Lucent Win, bus PCI,
  - Controller Twido: TWD LMDA 20DRT Versione 2.0,
  - Modem (collegato a Twido): tipo WESTERMO TD-33/V.90 codice di riferimento SR1 MOD01 disponibile nel nuovo catalogo Twido (settembre 03) (vedere l'*Appendice 2, p. 106*),
    Solo per i clienti del Nord America: In quest'area è disponibile il modem TD-33/V.90 US,

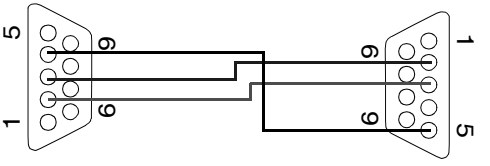
- Cavo: TSX PCX 1031 collegato alla porta 1 di comunicazione Twido e un adattatore: 9 pin maschio/9 pin maschio per incrociare Rx e Tx durante la connessione tra il modem Westermo e il controller Twido (vedere l'*Appendice 1, p. 105*). È possibile anche utilizzare il cavo TSX PCX 1130 (conversione RS485/232 e incrocio Rx/Tx).



Per questa verifica, utilizzare due linee telefoniche analogiche interne all'azienda, senza utilizzare il numero di telefono completo ma solo l'interno. È per questo che vi sono solo 4 cifre per il numero di telefono del modem Toshiba V.90 interno. Per questo test, i parametri di connessione (Twidosoft menu "Preferenze", "Gestione delle connessioni") erano impostati sui valori predefiniti, cioè timeout = 5000 e timeout interruzione = 20.

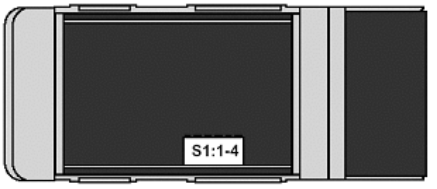
Appendice 1

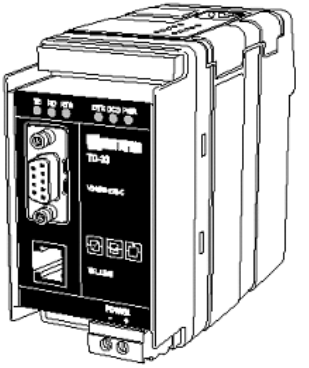
Adattatore per null modem per cavo TSX PCX 1031 e modem Westermo TD-33:



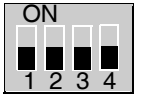
Appendice 2

Modem Westermo TD-33, codice di riferimento Schneider **SR1 MOD01**(1). Questo modem gestisce quattro microinterruttori DIP che devono tutti trovarsi su **OFF**:





Impostazioni di fabbrica



Usa configurazione memorizzata (velocità e formato, ecc)
Disabilita DTR Hotcall, Auto Band

Nota:
1. Alcuni prodotti potrebbero non essere compatibili e/o disponibili in tutte le aree. Per conoscere la disponibilità, contattare il distributore locale Schneider.

Appendice 3

Modem Wavecom WMOD2B, codice di riferimento Schneider **SR1 MOD02**(1) dual band (900/1800Hz):



Nota:
1. Alcuni prodotti potrebbero non essere compatibili e/o disponibili in tutte le aree. Per conoscere la disponibilità, contattare il distributore locale Schneider.

Appendice 4

Codici di riferimento dei prodotti utilizzati in questo documento:
- Prodotto Twido: TWD LMDA 20DRT,
- Software Twidosoft: TWD SPU 1002 V10M,
- Cavo TSX PCX 1031,
- Cavo TSX PCX 1130,
- Modem RTU: Westermo TD-33/V90 **SR1 MOD01**(1),
- Modem GSM: Wavecom WMOD2B **SR1 MOD02**(1).

Nota:
1. Alcuni prodotti potrebbero non essere compatibili e/o disponibili in tutte le aree. Per conoscere la disponibilità, contattare il distributore locale Schneider.

Comunicazioni in collegamento remoto

Introduzione

Il collegamento remoto è costituito da un bus master/slave ad alta velocità progettato per assicurare lo scambio di una piccola quantità di dati tra il controller master e un massimo di sette controller remoti (slave). I dati dell'applicazione o i dati di I/O vengono trasferiti in base alla configurazione dei controller remoti. È possibile combinare vari tipi di controller remoti. Alcuni possono essere controller di I/O remoti, altri controller di estensione.

Nota: il controller master contiene informazioni che riguardano l'indirizzo di un I/O remoto, ma non sa a quale specifico controller corrisponda questo indirizzo. Pertanto, il master non può confermare che tutti gli I/O remoti utilizzati nell'applicazione utente esistano veramente. Verificare con attenzione che questi I/O remoti siano realmente presenti.

Nota: il bus degli I/O remoti e il protocollo utilizzati sono proprietari e non è possibile inserire in rete dispositivi di terze parti.

⚠ ATTENZIONE

FUNZIONAMENTO NON PREVISTO DI APPARECCHIATURE

- Verificare che esista un unico controller master su un collegamento remoto e che ciascun slave abbia un indirizzo univoco. La mancata osservanza di questa precauzione può modificare i dati o produrre risultati imprevisti o ambigui.
- Accertarsi che tutti gli slave abbiano un indirizzo univoco. Due slave non possono avere lo stesso indirizzo. La mancata osservanza di questa precauzione può modificare i dati o produrre risultati imprevisti o ambigui.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

Nota: il collegamento remoto richiede una connessione EIA RS-485 e può essere eseguito solo su una porta di comunicazione alla volta.

Configurazione hardware

Il collegamento remoto deve utilizzare una porta EIA RS-485 con almeno 3 fili. È possibile configurarlo in modo da utilizzare la prima porta o la seconda porta opzionale.

Nota: è possibile configurare come collegamento remoto solo una porta di comunicazione alla volta.

La tabella seguente elenca le periferiche che possono essere utilizzate:

Controller	Porta	Specifiche tecniche
TWDLC•A10/16/24DRF, TWDLCA•40DRF, TWDLMDA20/40DUK, TWDLMDA20/40DTK, TWDLMDA20DRT	1	Controller di base dotato di una porta EIA 485 RS a 3 fili con connettore mini DIN.
TWDNOZ485D	2	Modulo di comunicazione dotato di una porta EIA 485 RS a 3 fili con connettore mini DIN. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNOZ485T	2	Modulo di comunicazione dotato di una porta EIA RS-485 a 3 fili con morsettiera. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNAC485D	2	Scheda di comunicazione dotata di una porta EIA RS-485 a 3 fili con connettore mini DIN. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.
TWDNAC485T	2	Scheda di comunicazione dotata di una porta EIA 485 RS a 3 fili con morsettiera. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.
TWDXCPODM	2	Modulo di espansione dotato di una porta EIA 485 RS a 3 fili con connettore mini DIN o di una porta EIA RS-485 a 3 fili con morsetto. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di comunicazione di espansione.

Nota: la verifica della presenza della porta 2 e della relativa configurazione (RS232 o RS485) è eseguita solo durante la messa sotto tensione o la reinizializzazione con il firmware del controller.

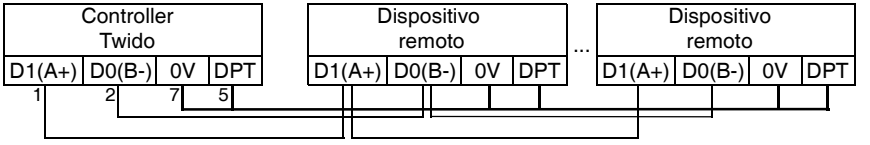
Collegamento dei cavi a ciascuna periferica

Nota: il segnale DPT sul pin 5 deve essere collegato allo 0V sul pin 7 per poter segnalare l'uso delle comunicazioni in collegamento remoto. Se questo segnale non è collegato alla messa a terra, il controller Twido master o slave passerà automaticamente in una modalità in cui vengono effettuati tentativi di comunicazione con TwidoSoft.

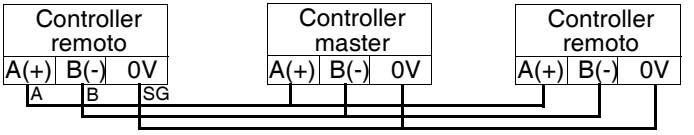
Nota: Il collegamento DPT a 0V è necessario solo se si è collegati a un controller della base sulla porta 1.

Le connessioni dei cavi a ciascuna periferica sono illustrate di seguito.

Connessione mini DIN



Connessione morsettiera



Configurazione software

Sul collegamento remoto deve essere definito un solo controller master. Inoltre, ciascun controller remoto deve mantenere un indirizzo slave univoco. Più master o slave che utilizzano gli stessi indirizzi possono danneggiare le trasmissioni o creare ambiguità.

> **⚠ ATTENZIONE**
>
> **FUNZIONAMENTO IMPREVISTO DELLE APPARECCHIATURE**
>
> Verificare che esista un unico controller master su un collegamento remoto e che ciascun slave abbia un indirizzo univoco. La mancata osservanza di questa precauzione può modificare i dati o produrre risultati imprevisti o ambigui.
>
> **La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.**

Configurazione del controller master

Il controller master viene configurato utilizzando TwidoSoft per gestire una rete di collegamenti remoti di un massimo di sette controller remoti. Questi sette controller remoti possono essere configurati come I/O remoti oppure come controller di estensione. L'indirizzo del master configurato utilizzando TwidoSoft corrisponde all'indirizzo 0.
Per configurare un controller master, occorre configurare con TwidoSoft la porta 1 o 2 a collegamento remoto e scegliere l'indirizzo 0 (Master).
Quindi, nella finestra "Aggiungi Controller remoto", determinare i controller slave, sia come I/O remoti, sia come controller di estensione nonché il loro indirizzo.

Configurazione del controller remoto

Per la configurazione di un controller master, occorre configurare con TwidoSoft la porta 1 o 2 a collegamento remoto e/o assegnare un indirizzo compreso tra 1 e 7. La tabella di seguito riepiloga le differenze e le restrizioni di ciascun tipo di configurazione di controller remoto:

Tipo	Programma applicativo	Accesso ai dati
I/O remoti	No Neanche una semplice istruzione "END" La modalità RUN dipende da quella del master.	%I e %Q Sul controller remoto è accessibile solo l'I/O locale (e non i relativi I/O di espansione).
Controller peer (di estensione)	Sì La modalità Run è indipendente da quella del master.	%INW e %QNW Da e verso ciascun controller di estensione è possibile trasmettere un massimo di 4 parole d'ingresso e 4 parole d'uscita

Sincronizzazione della scansione del controller remoto

Il ciclo di aggiornamento del collegamento remoto non è sincronizzato con la scansione del controller master. La comunicazione con i controller remoti è gestita tramite interrupt e avviene come task in background in parallelo con l'esecuzione della scansione del controller master. Alla fine del ciclo di scansione, i valori più aggiornati vengono letti nei dati dell'applicazione da utilizzare per la successiva esecuzione del programma. Questa elaborazione è uguale sia per i controller di I/O remoti sia per i controller di estensione.

Tutti i controller possono verificare l'attività del collegamento generale utilizzando il bit di sistema %S111 ma per ottenere la sincronizzazione, un master o un controller di estensione devono utilizzare il bit di sistema %S110. Questo bit viene impostato a 1 dopo che si è verificato un ciclo di aggiornamento completo. Il programma applicativo azzera il valore del bit.

Il master può abilitare o disabilitare il collegamento remoto utilizzando il bit di sistema %S112. I controller possono verificare la configurazione e il funzionamento corretti del collegamento remoto utilizzando %S113. Il segnale DPT sulla porta 1, utilizzato per stabilire se TwidoSoft è collegato, viene rilevato e riportato su %S100. La tabella seguente riepiloga questi dati:

Bit di sistema	Stato	Indicazione
%S100	0	master/slave: DPT non attivo (il cavo TwidoSoft NON è collegato)
	1	master/slave: DPT attivo (il cavo TwidoSoft è collegato)
%S110	0	master/slave: azzerato dall'applicazione
	1	master: tutti gli scambi a collegamento remoto sono stati completati (solo I/O remoti) slave: scambio con il master completato
%S111	0	master: scambio di collegamento remoto singolo eseguito slave: scambio di collegamento remoto singolo rilevato
	1	master: scambio di collegamento remoto singolo in corso slave: scambio di collegamento remoto singolo rilevato
%S112	0	master: collegamento remoto disabilitato
	1	master: collegamento remoto abilitato
%S113	0	master/slave: configurazione/funzionamento collegamento remoto corretta
	1	master: configurazione/funzionamento collegamento remoto errata slave: funzionamento collegamento remoto errato

Riavvio del controller master

Se un controller master viene riavviato, si verifica uno dei seguenti eventi:

- Un avvio a freddo (%S0 = 1) forza la reinizializzazione delle comunicazioni.
- Un avvio a caldo (%S1 = 1) forza la reinizializzazione delle comunicazioni.
- In modalità Stop, il master continua a comunicare con gli slave.

Riavvio del controller slave

Se un controller slave viene riavviato, si verifica uno dei seguenti eventi:

- Un avvio a freddo (%S0 = 1) forza la reinizializzazione delle comunicazioni.
- Un avvio a caldo (%S1 = 1) forza la reinizializzazione delle comunicazioni.
- In modalità Stop, lo slave continua a comunicare con il master. Se il master è in stato di Stop:
  - Agli I/O remoti viene applicato lo stato di Stop.
  - Il controller di estensione continua nello stato corrente.

Stop del controller master

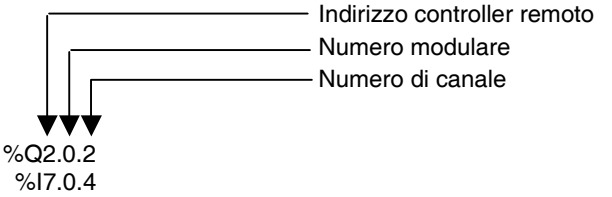
Quando il controller master passa in modalità Stop, tutte le periferiche slave continuano a comunicare con il master. Quando il master indica la richiesta di uno Stop, il controller I/O remoto entra in Stop ma i controller di estensione continuano nello stato Run o Stop corrente.

Accesso ai dati I/O remoti

Il controller remoto configurato come I/O remoto non possiede né esegue il proprio programma applicativo. Gli ingressi e le uscite digitali base del controller remoto sono una semplice estensione di quelli del controller master. L'applicazione deve usare solo il meccanismo di indirizzamento a tre cifre completo fornito.

Nota: il numero di modulo è sempre zero per gli I/O remoti.

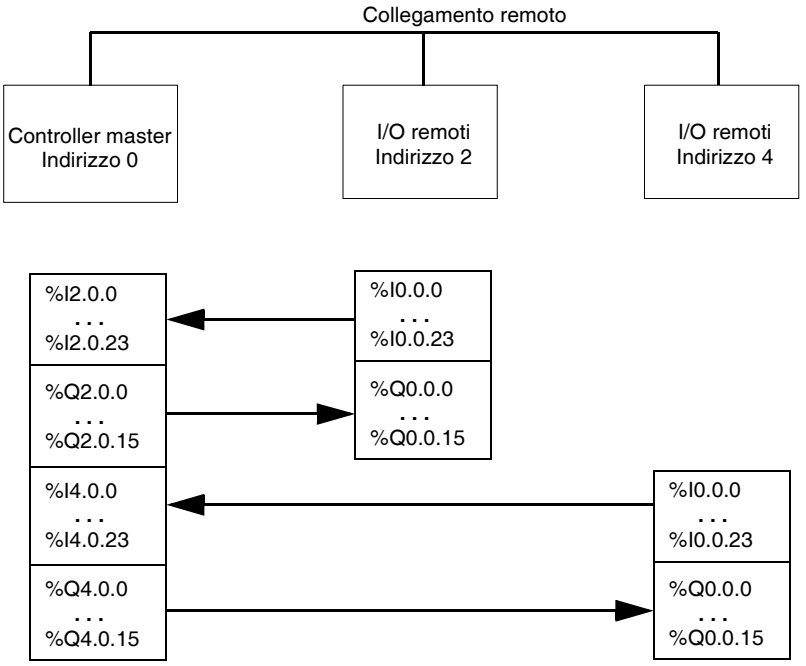
Illustrazione



Per la comunicazione con gli I/O remoti, il controller master utilizza la classificazione standard di ingresso e uscita %I e %Q. Per accedere al terzo bit di uscita dell'I/O remoto configurato all'indirizzo 2, si utilizza l'istruzione %Q2.0.2. Analogamente, per leggere il quinto bit di ingresso degli I/O remoti configurati all'indirizzo 7, si utilizza l'istruzione %I7.0.4.

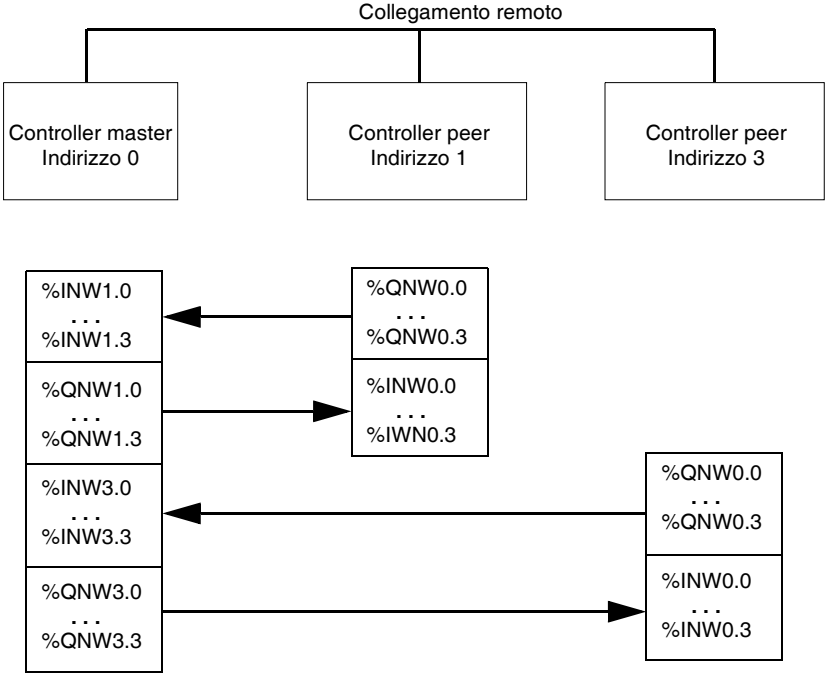
Nota: l'accesso del master è limitato all'I/O digitale che fa parte dell'I/O locale del remoto. Non è possibile trasferire alcun I/O analogico o di espansione, a meno che non si utilizzino comunicazioni peer.

Illustrazione



Accesso ai dati del controller di estensione

Per comunicare con controller di estensione, il master utilizza le parole di rete %INW e %QNW per scambiare i dati. Per accedere ad ogni controller di estensione sulla rete si utilizza il relativo indirizzo remoto "j" utilizzando le parole %INWj.k e %QNWj.k. Ciascun controller di estensione sulla rete utilizza le parole da %INW0.0 a %INW0.3 e da %QNW0.0 a %QNW0.3 per accedere ai dati sul master. Le parole di rete sono aggiornate automaticamente quando i controller si trovano in modalità Run o Stop.
L'esempio che segue illustra lo scambio di un master con due controller di estensione configurati.



Il collegamento remoto non prevede lo scambio di messaggi peer-to-peer. È possibile utilizzare i programmi d'applicazione del master per gestire le parole di rete per trasferire le informazioni tra i controller remoti utilizzando il master come bridge.

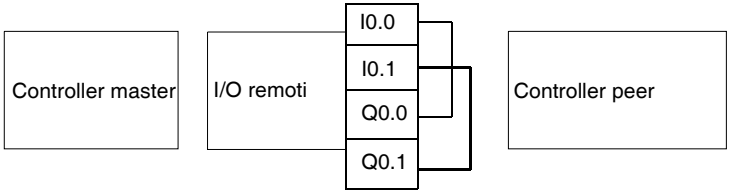
Informazioni di stato

Oltre ai bit di sistema descritti in precedenza, il master mantiene lo stato sulla presenza e sulla configurazione dei controller remoti. Ciò avviene nelle parole di sistema. Il controller remoto o il controller master possono ottenere il valore dell'ultimo errore verificatosi durante la comunicazione sul collegamento remoto nella parola di sistema %SW112.

Parole di sistema	Utilizzo	
%SW111	Stato del collegamento remoto: due bit per ciascun controller remoto (solo master)	
	x0-6	0 - Controller remoto 1-7 non presente
		1 - Controller remoto 1-7 presente
	x8-14	0 - I/O remoti rilevati sul controller remoto 1-7
		1 – Controller di estensione rilevato sul controller remoto 1-7
%SW112	Codice di errore per configurazione/funzionamento collegamento remoto	
		0 – Operazioni riuscite
		1 – Timeout scaduto (slave)
		2 – Rilevato errore di checksum (slave)
		3 – Mancata corrispondenza della configurazione (slave)
%SW113	Configurazione del collegamento remoto: due bit per ciascun controller remoto (solo master)	
	x0-6	0 - Controller remoto 1-7 non configurato
		1 - Controller remoto 1-7 configurato
	x8-14	0 - I/O remoti configurati come controller remoto 1-7
		1 - Controller di estensione configurato come controller remoto 1-7

Esempio di collegamento remoto

Per configurare un collegamento remoto è necessario:
1. Configurare l'hardware.
2. Collegare i controller.
3. Collegare il cavo di comunicazione tra il PC e i controller.
4. Configurare il software.
5. Scrivere un'applicazione.
I diagrammi che seguono spiegano l'uso del collegamento remoto con gli I/O remoti e un controller di estensione.
Fase 1: Configurare l'hardware:

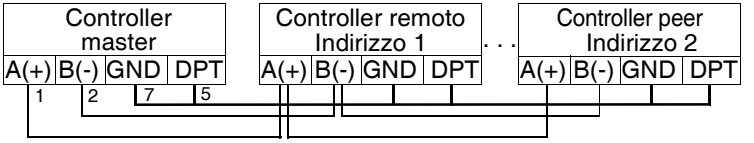


La configurazione hardware è costituita da tre controller di base di qualsiasi tipo. La porta 1 viene utilizzata in base a due tipi di comunicazione. Una modalità consiste nel configurare e trasferire il programma applicativo con TwidoSoft. La seconda modalità è destinata alla rete di collegamenti remoti. Se disponibile, è possibile utilizzare una porta 2 opzionale su uno dei controller, ma ogni controller supporta solo un collegamento remoto singolo.

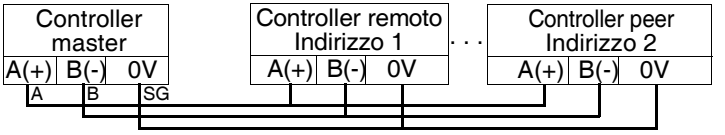
Nota: In questo esempio, i primi due ingressi sugli I/O remoti sono cablati sulle due prime uscite.

Fase 2: Cablaggi controller

Connessione mini DIN

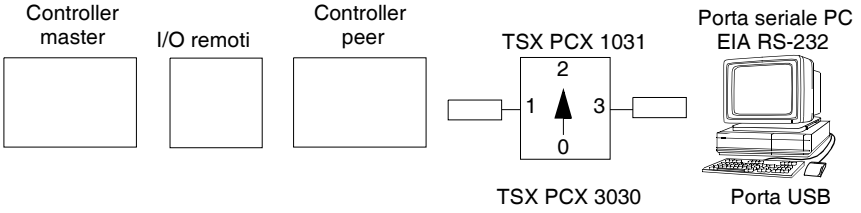


Connessione morsettiera



Collegare insieme i cavi di segnale A(+) e B(-). Su ciascun controller, il segnale DPT è collegato a terra. Anche se il collegamento a terra del segnale non è richiesto per l'uso con un collegamento remoto sulla porta 2 (cartuccia o modulo di comunicazione opzionale), adottarlo è una buona prassi.

Fase 3: Collegare il cavo di comunicazione tra il PC e i controller:



Il cavo di programmazione multifunzione TSX PCX 1031 o TSX PCX 3030 viene utilizzato per comunicare con ciascuno dei tre controller di base. Verificare che il cavo si trovi sulla posizione 2 dell'interruttore. Per programmare ciascun controller, è necessario predisporre una comunicazione punto-punto con ogni controller. Per stabilire tale comunicazione: effettuare il collegamento alla porta 1 del primo controller, trasferire i dati di configurazione e dell'applicazione e impostare il controller sullo stato Run. Ripetere la procedura per ciascun controller.

Nota: Il cavo deve essere spostato dopo ogni configurazione del controller e trasferimento dell'applicazione.

Fase 4: Configurazione del software:
Ciascuno dei tre controller utilizza TwidoSoft per creare una configurazione e, se necessario, il programma applicativo.
Per il controller master, modificare la configurazione di comunicazione del controller per impostare il protocollo su "collegamento remoto" e l'indirizzo su "0 (Master)".

Impostazione delle comunic. del controller
Tipo: Collegamento remoto
Indirizzo: 0 (Master)

Configurare il controller remoto sul master aggiungendo un "I/O remoto" all'indirizzo "1" e un "Controller di estensione" all'indirizzo "2".

Aggiungi controller remoti

Uso controller: I/O remoti
Indirizzo remoto: 1

Uso controller: controller peer
Indirizzo remoto: 2

Per il controller configurato come I/O remoto, verificare che la configurazione di comunicazione del controller sia impostata su "Collegamento remoto" e l'indirizzo sia impostato a "1".

Impostazione delle comunic. del controller
Tipo: Collegamento remoto
Indirizzo: 1

Per il controller configurato come estensione, verificare che la configurazione di comunicazione del controller sia impostata su "Collegamento remoto" e l'indirizzo sia impostato a "2".

Impostazione delle comunic. del controller
Tipo: Collegamento remoto
Indirizzo: 2

Fase 5: Scrittura delle applicazioni:
Per il controller master, scrivere il seguente programma applicativo:

```
LD 1

[%MW0 := %MW0 +1]
[%QNW2.0 := %MW0]
[%MW1 := %INW2.0]

LD %I0.0
ST %Q1.00.0
LD %I1.0.0
ST %Q0.0

LD %I0.1
ST %Q1.0.1
LD %I1.0.1
ST %Q0.1
```

Per il controller configurato come I/O remoto, non scrivere alcun programma applicativo.
Per il controller configurato come estensione, scrivere la seguente applicazione:

```
LD 1
[%QNW0.0 := %INW0.0]
```

In questo esempio, l'applicazione master incrementa una parola di memoria interna e la comunica al controller di estensione utilizzando una sola parola di rete. Il controller di estensione assume la parola ricevuta dal master e la ritrasmette. Nel master, questa trasmissione viene ricevuta e memorizzata da una parola di memoria diversa.
Per la comunicazione con il controller I/O remoto, il master invia gli ingressi locali alle uscite dell'I/O remoto. Con il cablaggio di I/O esterno dell'I/O remoto, i segnali vengono restituiti e ripristinati dal master.

Comunicazioni ASCII

Introduzione

Il protocollo ASCII fornisce ai controller Twido un semplice protocollo di modalità carattere half-duplex per trasmettere e/o ricevere dati con una sola periferica. Questo protocollo è supportato utilizzando l'istruzione EXCHx e gestito tramite il blocco funzione %MSGx.
Quando si utilizza il protocollo ASCII, sono disponibili i tre tipi di comunicazione seguenti:

- Sola trasmissione
- Trasmissione/Ricezione
- Sola ricezione

La dimensione massima dei pacchetti di dati trasmessi e/o ricevuti utilizzando l'istruzione EXCHx è di 256 byte.

Configurazione hardware

Un collegamento ASCII (vedere i bit di sistema %S103 e %S104 (Vedi *Bit di sistema (%S), p. 604*)) può essere stabilito sulla porta EIA RS-232 o EIA RS-485 e può essere eseguito al massimo su due porte di comunicazione alla volta.
La tabella seguente elenca le periferiche che possono essere utilizzate:

Controller	Porta	Specifiche tecniche
TWDLC•A10/16/24DRF, TWDLCA•40DRF, TWDLMDA20/40DTK, TWDLMDA20DRT	1	Controller di base dotato di una porta EIA 485 RS a 3 fili con connettore mini DIN.
TWDNOZ232D	2	Modulo di comunicazione dotato di una porta EIA RS-232 a 3 fili con connettore mini DIN. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNOZ485D	2	Modulo di comunicazione dotato di una porta EIA 485 RS a 3 fili con connettore mini DIN. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNOZ485T	2	Modulo di comunicazione dotato di una porta EIA RS-485 a 3 fili con morsettiera. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNAC232D	2	Scheda di comunicazione dotata di una porta EIA RS-232 a 3 fili con connettore mini DIN. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.

Controller	Porta	Specifiche tecniche
TWDNAC485D	2	Scheda di comunicazione dotata di una porta EIA RS-485 a 3 fili con connettore mini DIN. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.
TWDNAC485T	2	Scheda di comunicazione dotata di una porta EIA 485 RS a 3 fili con morsettiera. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.
TWDXCPODM	2	Modulo di espansione del pannello visualizzatore dotato di una porta EIA RS-232 a tre fili con un connettore mini DIN, di una porta EIA RS-485 a 3 fili con un connettore mini DIN e di una porta EIA RS-485 a 3 fili con una morsettiera. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di comunicazione di espansione.

Nota: la verifica della presenza della porta 2 e della relativa configurazione (RS232 o RS485) viene eseguita solo durante la messa sotto tensione o la reinizializzazione con il firmware del controller.

Cablaggio nominale

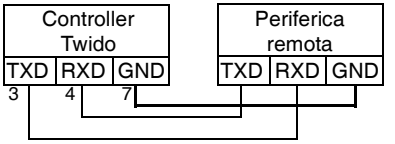
Di seguito sono illustrate le connessioni del cablaggio nominale per le porte di tipo EIA RS-232 e EIA RS-485.

Nota: Se si utilizza la porta 1 sul controller Twido, il segnale DPT sul pin 5 deve essere collegato a 0V sul pin 7. Per il controller Twido, questo significa che le comunicazioni mediante la porta 1 sono in formato ASCII e non nel protocollo utilizzato per comunicare con il software TwidoSoft.

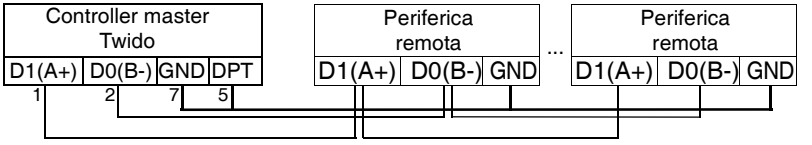
Le connessioni dei cavi a ciascuna periferica sono illustrate di seguito.

Connessione mini DIN

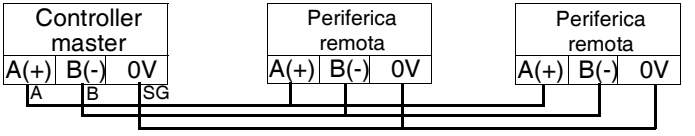
Cavo EIA RS-232



Cavo EIA RS-485



Connessione morsettiera



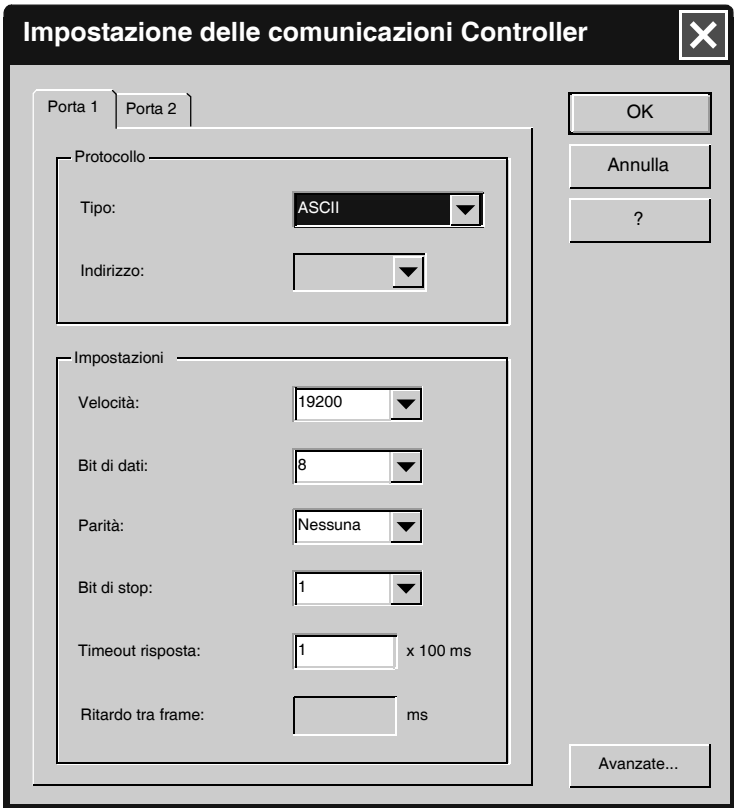
Configurazione software

Per configurare il controller in modo che utilizzi una connessione seriale per inviare e ricevere caratteri utilizzando il protocollo ASCII, è necessario:

Passo	Descrizione
1	Configurare la porta seriale per il protocollo ASCII utilizzando TwidoSoft.
2	Creare nell'applicazione una tabella di trasmissione/ricezione utilizzata dall'istruzione EXCHx.

Configurazione della porta

Un controller Twido può utilizzare la propria porta principale 1 o una porta 2 configurata opzionale per il protocollo ASCII. Per configurare una porta seriale per il protocollo ASCII:

Passo	Azione
1	Definire tutti i moduli o le schede di comunicazione supplementari configurati sulla base.
2	Nel browser dell'applicazione, fare clic con il pulsante destro del mouse sulla porta e selezionare **Modifica impostazioni di Com. del Controller...** **Risultato:** viene visualizzata la seguente finestra. 
3	Selezionare il tipo di porta seriale ASCII nella casella di riepilogo **Tipo protocollo**.
4	Impostare i parametri di comunicazione associati.
5	Fare clic sul pulsante **Avanzate** per impostare i parametri avanzati.

Configurazione della tabella di trasmissione e ricezione per la modalità ASCII

Le dimensioni massime dei pacchetti dati trasmessi o ricevuti sono di 256 byte. La tabella delle parole associata all'istruzione EXCHx è composta dalle tabelle di controllo di trasmissione e ricezione.

	Byte più significativo	**Byte meno significativo**
Tabella di controllo	Comando	Lunghezza (trasmissione/ricezione)
	Riservato (0)	Riservato (0)
Tabella di trasmissione	Trasmesso byte 1	Trasmesso byte 2
	...	...
		Trasmesso byte n
	Trasmesso byte n+1	
Tabella di ricezione	Ricevuto byte 1	Ricevuto byte 2
	...	...
		Ricevuto byte p
	Ricevuto byte p+1	

Tabella di controllo

Il byte **Lunghezza** contiene la lunghezza della tabella di trasmissione in byte (250 max), che viene sovrascritta dal numero di caratteri ricevuti alla fine della ricezione, se questa è richiesta.
Il byte **Comando** deve contenere uno dei seguenti elementi:

- 0: Sola trasmissione
- 1: Trasmissione/Ricezione
- 2: Sola ricezione

Tabelle di trasmissione/ ricezione

In modalità di sola trasmissione, le tabelle di controllo e trasmissione vengono completate prima dell'esecuzione dell'istruzione EXCHx e possono essere di tipo %KW o %MW. Nessuno spazio è richiesto per la ricezione dei caratteri in modalità di sola trasmissione. Dopo aver trasmesso tutti i byte, %MSGx.D viene impostato a 1 ed è possibile eseguire una nuova istruzione EXCHx.
In modalità di trasmissione/ricezione, le tabelle di controllo e trasmissione vengono completate prima dell'esecuzione dell'istruzione EXCHx e devono essere di tipo %MW. Al termine della tabella di trasmissione sono richiesti fino a 256 byte di spazio per la ricezione. Una volta che sono stati trasmessi tutti i byte, il controller Twido passa alla modalità di ricezione e attende di ricevere i byte.
In modalità di sola ricezione, la tabella di controllo viene completata prima dell'esecuzione dell'istruzione EXCHx e deve essere di tipo %MW. Al termine della tabella di controllo sono richiesti fino a 256 byte di spazio per la ricezione. Il controller Twido entra immediatamente in modalità di ricezione e attende di ricevere i byte.
La ricezione si conclude quando viene ricevuto il byte di chiusura del pacchetto dati o quando la tabella di ricezione è piena. In questo caso compare un errore di overflow della tabella di ricezione nelle parole %SW63 e %SW64. Se viene configurato un timeout diverso da zero, la ricezione si conclude al completamento del timeout. Se si seleziona un valore di timeout uguale a zero, non vi sarà alcun timeout di ricezione. Quindi, per interrompere la ricezione, occorre attivare l'ingresso %MSGx.R.

Scambio di messaggi

Per la comunicazione, il linguaggio offre due funzionalità:
- **Istruzione EXCHx:** per trasmettere/ricevere messaggi.
- **Blocco funzione %MSGx:** per controllare gli scambi di messaggi.

Il controller Twido utilizza il protocollo configurato per la porta specifica durante l'elaborazione di un'istruzione EXCHx.

Nota: ogni porta di comunicazione può essere configurata per protocolli diversi o per lo stesso protocollo. È possibile accedere all'istruzione EXCHx o al blocco funzione %MSGx per ciascuna porta di comunicazione aggiungendo il numero di porta (1 o 2).

Istruzione EXCHx

L'istruzione EXCHx consente a un controller Twido di inviare e/o ricevere informazioni da periferiche ASCII. L'utente definisce una tabella di parole (%MWi:L o %KWi:L) contenente le informazioni di controllo oltre ai dati da inviare e/o ricevere (fino a 256 byte in trasmissione e/o ricezione). Il formato della tabella di parole è stato descritto in precedenza.
Uno scambio di messaggi viene eseguito utilizzando l'istruzione EXCHx.

Sintassi: [EXCHx %MWi:L]

dove: x = numero della porta (1 o 2)

L = numero di parole della tabella delle parole di comando della tabella di trasmissione e ricezione

Il controller Twido deve concludere lo scambio relativo alla prima istruzione EXCHx prima di poter avviare una seconda istruzione di scambio. Nel caso di invio di più messaggi, utilizzare il blocco funzione %MSGx.
L'elaborazione dell'istruzione tramite lista EXCHx avviene immediatamente, con qualsiasi trasmissione avviata dietro il controllo dell'interrupt (anche la ricezione dei dati è controllata dall'interrupt), e questa viene considerata elaborazione in background.

Blocco funzione %MSGx

L'utilizzo del blocco funzione %MSGx è opzionale, può essere impiegato per gestire gli scambi di dati. Il blocco funzione %MSGx ha tre obiettivi:

- **Controllo degli errori di comunicazione**
  La ricerca degli errori consente di verificare che il parametro L (lunghezza della tabella di parole) programmato con l'istruzione EXCHx sia sufficientemente ampio da contenere il messaggio da inviare. Questo viene confrontato con la lunghezza programmata nel byte meno significativo della prima parola della tabella di parole.
- **Coordinamento di più messaggi**
  Per garantire il coordinamento quando si inviano più messaggi, il blocco funzione %MSGx fornisce le informazioni richieste per determinare il momento in cui viene completata la trasmissione del messaggio precedente.
- **Trasmissione di messaggi di priorità**
  Il blocco funzione %MSGx consente di interrompere la trasmissione del messaggio corrente per consentire l'invio immediato di un messaggio urgente.

Il blocco funzione %MSGx dispone di un ingresso e due uscite associati:

I/O	Definizione	Descrizione
R	Azzera ingresso	Impostato a 1: reinizializza la comunicazione o azzera il blocco (%MSGx.E = 0 e %MSGx.D = 1).
%MSGx.D	Comunicazione completata	0: richiesta in corso. 1: comunicazione eseguita se viene ricevuto il segnale di fine trasmissione, carattere finale, errore o azzeramento del blocco.
%MSGx.E	errore	0: lunghezza messaggio corretta e collegamento corretto. 1: se il comando è sbagliato, se la tabella è configurata in modo non corretto, se viene ricevuto un carattere errato (velocità, parità, ecc.) o se la tabella di ricezione è piena.

Limiti

È importante tenere presenti le seguenti limitazioni:
- La disponibilità e il tipo della porta 2 (vedere %SW7) vengono controllati solo all'accensione o al reset.
- L'elaborazione dei messaggi sulla porta 1 viene terminata al collegamento di TwidoSoft.
- EXCHx o %MSG non possono essere elaborati su una porta configurata come collegamento remoto.
- EXCHx termina l'elaborazione Modbus slave attivo.
- L'elaborazione delle istruzioni EXCHx non viene ritentata in caso di errore.
- L'ingresso Reset (R) può essere utilizzato per terminare l'elaborazione in ricezione di un'istruzione EXCHx.
- Le istruzioni EXCHx possono essere configurate con un timeout per terminare la ricezione.
- Più messaggi sono controllati tramite %MSGx.D.

Errori e condizioni di funzionamento

Se si verifica un errore utilizzando un'istruzione EXCHx, i bit %MSGx.D e %MSGx.E vengono impostati a 1, la parola di sistema %SW63 contiene il codice di errore della porta 1 e %SW64 contiene il codice di errore della porta 2.

Parole di sistema	Utilizzo
%SW63	Codice di errore EXCH1: 0 – Operazione riuscita 1 - Numero di byte da emettere troppo grande (> 250) 2 - Tabella di emissione troppo piccola 3 - Tabella di parole troppo piccola 4 - Overflow tabella di ricezione 5 - Timeout scaduto 6 - Errore di trasmissione 7 - Comando errato nella tabella 8 - Porta selezionata non configurata/disponibile 9 – Errore di ricezione 10 - Impossibile utilizzare %KW se in ricezione 11 - Offset di trasmissione più grande della tabella di trasmissione 12 - Offset di ricezione più grande della tabella di ricezione 13 - Interruzione dell'elaborazione di EXCH da controller
%SW64	Codice d'errore EXCH2: Vedere %SW63.

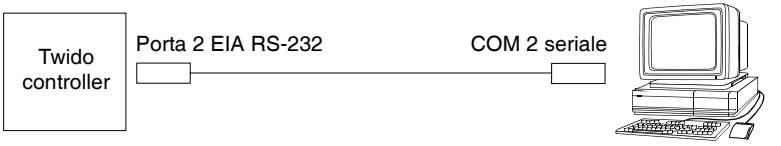
Conseguenze del riavvio del controller sulla comunicazione

Se un controller viene riavviato, si verifica uno dei seguenti eventi:
- Un avvio a freddo (%S0 = 1) forza la reinizializzazione delle comunicazioni.
- Un avvio a caldo (%S1 = 1) forza la reinizializzazione delle comunicazioni.
- In modalità Stop, il controller interrompe tutte le comunicazioni ASCII.

Esempio di collegamento ASCII

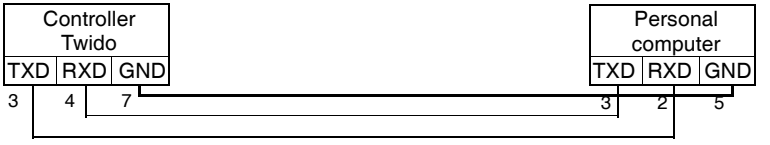
Per configurare un collegamento ASCII è necessario:
1. Configurare l'hardware.
2. Collegare il cavo di comunicazione ASCII.
3. Configurare la porta.
4. Scrivere un'applicazione.
5. Inizializzare l'editor delle tabelle di animazione.
Il diagramma seguente illustra l'uso delle comunicazioni ASCII con un emulatore di terminale su un PC.

Fase 1: Configurazione hardware:



La configurazione hardware è costituita da due connessioni seriali dal PC al controller Twido con una porta EIA RS-232 2 opzionale. Su un controller modulare, la porta 2 opzionale è di tipo TWDNOZ232D o TWDNAC232D nel TWDXCPODM. Su un controller compatto, la porta 2 opzionale è di tipo TWDNAC232D.
Per configurare il controller, collegare il cavo TSXPCX1031 (non illustrato) alla porta 1 del controller Twido. Quindi collegare il cavo alla porta COM 1 sul PC. Verificare che l'interruttore sia in posizione 2. Infine, collegare la porta COM 2 del PC alla porta 2 EIA RS-232 opzionale sul controller Twido. Lo schema di cablaggio è illustrato nella fase successiva.

Fase 2: Schema del cavo di comunicazione ASCII (EIA RS-232):



Il numero minimo di fili utilizzati nel cavo di comunicazione ASCII è 3. Incrociare i segnali di trasmissione e ricezione.

Nota: all'estremità PC del cavo, l'handshaking può richiedere connessioni supplementari (ad esempio, Data Terminal Ready e Data Set Ready). Nessuna connessione supplementare è richiesta dal controller Twido.

Fase 3: Configurazione della porta:

Hardware -> Aggiungi opzione TWDNOZ232D	
Porta seriale 2	
Protocollo	ASCII
Indirizzo	
Velocità di trasmissione	19200
Bit di dati	8
Parità	Nessuna
Bit di stop	1
Timeout risposta (x100ms)	100
Ritardo tra frame (ms)	
Carattere iniziale	
1° carattere finale	65
2° carattere finale	
Stop senza ricezione (ms)	
Stop sul numero di byte ricevuti	

Emulatore di terminale su PC	
Porta:	COM2
Velocità:	19200
Dati:	8 bit
Parità:	Nessuna
Arresto:	1 bit
Controllo di flusso:	Nessuno

Utilizzare una semplice applicazione di emulazione del terminale sul PC per configurare la porta 2 e verificare che non vi sia controllo del flusso.
Utilizzare TwidoSoft per configurare la porta del controller. Innanzi tutto, configurare l'opzione hardware. In questo esempio, viene aggiunta la porta TWDNOZ232D al controller modulare di base.
Quindi inizializzare la configurazione della comunicazione del controller con le stesse impostazioni dell'emulatore del terminale sul PC. In questo esempio, la lettera "A" maiuscola viene scelta per il primo carattere finale per concludere la ricezione dei caratteri. Viene selezionato un timeout di 10 secondi per il parametro "Timeout risposta". Solo uno di questi due parametri sarà utilizzato, a seconda di quale delle due situazioni si verifica per prima.

Fase 4: Scrittura di un'applicazione:

```
LD 1
[%MW10 := 16#0104 ]
[%MW11 := 16#0000 ]
[%MW12 := 16#4F4B ]
[%MW13 := 16#0A0D ]
LD 1
AND %MSG2.D
[EXCH2 %MW10:8]
LD %MSG2.E
ST %Q0.0
END
```

Utilizzare TwidoSoft per creare un programma applicativo con tre parti principali. Primo, inizializzare le tabelle di controllo e di trasmissione da utilizzare per l'istruzione EXCH. In questo esempio, viene impostato un comando sia per inviare sia per ricevere i dati. La quantità di dati da inviare è impostata a 4 byte, come definito nell'applicazione, seguito dal carattere di chiusura del pacchetto dati utilizzato, in questo caso il primo carattere finale "A". Nella tabella di animazione, i caratteri iniziali e finali non vengono visualizzati ma sono riportati solo i caratteri dei dati. Questi caratteri, tuttavia, quando utilizzati vengono trasmessi o verificati automaticamente alla ricezione, da %SW63 e %SW64.
Quindi controllare il bit di stato associato a %MSG2 ed eseguire l'istruzione EXCH2 solo se la porta è pronta. Per l'istruzione EXCH2, viene specificato un valore di 8 parole. Esistono 2 parole di controllo (%MW10 e %MW11), 2 parole da utilizzare per le informazioni di trasmissione (%MW12 e %MW13) e 4 parole per ricevere i dati (da %MW14 a %MW16).
Al termine lo stato di errore di %MSG2 viene rilevato e memorizzato nel primo bit di uscita sull'I/O del controller base locale. Per aumentare la precisione, è possibile effettuare ulteriori controlli usando %SW64.

Fase 5: Inizializzazione dell'editor delle tabelle di animazione:

Indirizzo	Corrente	Formato ritenuto
1 %MW10	0104	Esadecimale
2 %MW11	0000	Esadecimale
3 %MW12	4F4B	Esadecimale
4 %MW13	0A0D	Esadecimale
5 %MW14	TW	ASCII
6 %MW15	ID	ASCII
7 %MW16	O	ASCII

Il passo finale consiste nello scaricare questa applicazione per controller e nell'eseguirla. Inizializzare l'editor delle tabelle di animazione per l'animazione e la visualizzazione delle parole da %MW10 a %MW16. Sull'emulatore di terminale, è possibile visualizzare i caratteri "O- K - CR - LF - A" fino alla scadenza del timeout di risposta del blocco EXCH. Sull'emulatore del terminale, digitare "T - W - I - D - O - A". Queste informazioni vengono scambiate con il controller Twido e visualizzate nell'editor delle tabelle di animazione.

Comunicazioni Modbus

Introduzione

Il protocollo Modbus è un protocollo master/slave che consente a un solo master di richiedere risposte agli slave o di agire in base alla richiesta. Il master può rivolgersi a un singolo slave o può trasmettere un messaggio in diffusione broadcast a tutti gli slave. Gli slave rispondono con un messaggio (risposta) alle richieste che vengono inviate loro singolarmente. Non vengono rinviate le risposte alle query provenienti da un master.

⚠ ATTENZIONE

FUNZIONAMENTO NON PREVISTO DI APPARECCHIATURE

- Verificare che esista un unico controller master Modbus sul bus e che ciascun slave Modbus abbia un indirizzo univoco. La mancata osservanza di questa precauzione può modificare i dati o produrre risultati imprevisti o ambigui.
- Accertarsi che tutti gli slave Modbus abbiano un indirizzo univoco. Due slave non possono avere lo stesso indirizzo. La mancata osservanza di questa precauzione può modificare i dati o produrre risultati imprevisti o ambigui.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

Configurazione hardware

Un collegamento Modbus può essere stabilito sulla porta EIA RS-232 or EIA RS-485 e può essere eseguito contemporaneamente su due porte di comunicazione alla volta. Ogni porta può ottenere il proprio indirizzo Modbus, utilizzando il bit di sistema %S101 e le parole di sistema %SW101 e %SW102 (Vedi *Bit di sistema (%S), p. 604*). . (Vedi anche *Parole di sistema (%SW), p. 614*)
La tabella seguente elenca le periferiche che possono essere utilizzate:

Dispositivo remoto	Porta	Specifiche tecniche
TWDLC•A10/16/24DRF, TWDLCA•40DRF, TWDLMDA20/40DTK, TWDLMDA20DRT	1	Controller di base dotato di una porta EIA 485 RS a 3 fili con connettore mini DIN.
TWDNOZ232D	2	Modulo di comunicazione dotato di una porta EIA RS-232 a 3 fili con connettore mini DIN. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNOZ485D	2	Modulo di comunicazione dotato di una porta EIA 485 RS a 3 fili con connettore mini DIN. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNOZ485T	2	Modulo di comunicazione dotato di una porta EIA RS-485 a 3 fili con morsettiera. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di espansione del pannello visualizzatore.
TWDNAC232D	2	Scheda di comunicazione dotata di una porta EIA RS-232 a 3 fili con connettore mini DIN. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.
TWDNAC485D	2	Scheda di comunicazione dotata di una porta EIA RS-485 a 3 fili con connettore mini DIN. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.
TWDNAC485T	2	Scheda di comunicazione dotata di una porta EIA RS-485 a 3 fili con connettore terminale. **Nota:** questa scheda è disponibile solo per i controller compatti a 16, 24 e 40 I/O e per il modulo di espansione del pannello visualizzatore.
TWDXCPODM	2	Modulo di espansione del pannello visualizzatore dotato di una porta EIA RS-232 a tre fili con un connettore mini DIN, di una porta EIA RS-485 a 3 fili con un connettore mini DIN e di una porta EIA RS-485 a 3 fili con una morsettiera. **Nota:** questo modulo è disponibile solo per i controller modulari. Se questo modulo è collegato, il controller non può avere un modulo di comunicazione di espansione.

Nota: la verifica della presenza della porta 2 e della relativa configurazione (RS232 o RS485) viene realizzata durante la messa sotto tensione o la reinizializzazione con il firmware del controller.

Cablaggio nominale

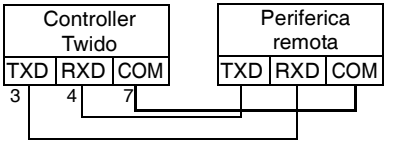
Di seguito sono illustrate le connessioni del cablaggio nominale per le porte di tipo EIA RS-232 e EIA RS-485.

Nota: se si utilizza la porta 1 sul controller Twido, è necessario collegare il segnale DPT sul pin 5 al circuito comune (COM) sul pin 7. Per il controller Twido, ciò significa che le comunicazioni mediante la porta 1 sono in formato Modbus e non nel protocollo utilizzato per comunicare con il software TwidoSoft.

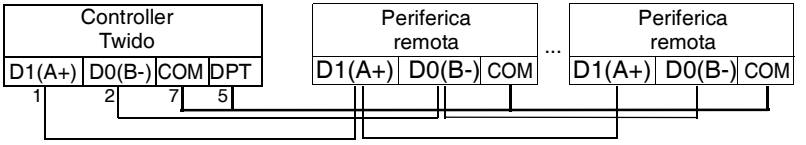
Di seguito sono descritte le connessioni dei cavi a ciascuna periferica.

Connessione mini DIN

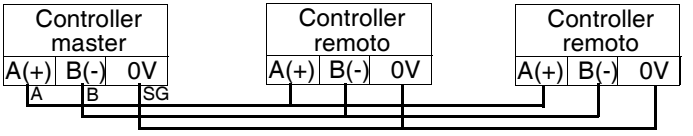
Cavo EIA RS-232



Cavo EIA RS-485



Connessione morsettiera



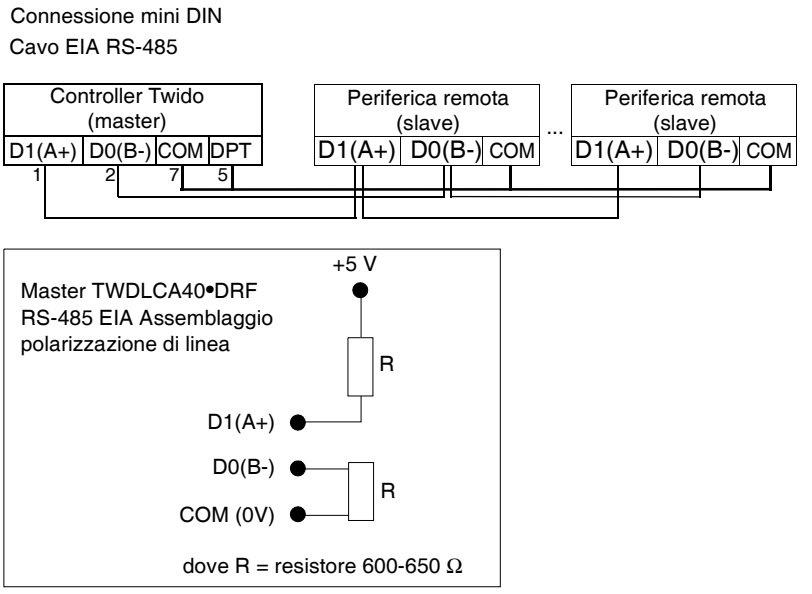
Polarizzazione di linea EIA RS-485 sui controller TWDLCA•40DRF

Non esiste una prepolarizzazione interna nei controller TWDLCA•40DRF. Quando si collega il controller master Modbus TWDLCA•40DRF alla rete Modbus EIA-485, pertanto, è necessaria la polarizzazione di linea esterna.
Quando non vi è attività di dati sulla coppia bilanciata EIA-485, le linee non sono utilizzate e quindi non sono soggette a disturbi o interferenze esterne. Per fare in modo che il ricevitore si mantenga in uno stato costante quando non è presente il segnale dati, il dispositivo master Modbus deve eseguire il bias della rete tramite la polarizzazione di linea esterna.

Nota: la polarizzazione di linea esterna EIA RS-485 deve essere implementata solo sul controller master Modbus. La polarizzazione non deve essere implementata su un qualunque dispositivo slave.

L'assieme della polarizzazione di linea esterna su TWDLCA•40DRF mini-DIN linea RS-485 EIA è costituito da:
- Un resistore pull-up tensione 5V su circuito D1(A+),
- Un resistore pull-down per il circuito comune sul circuito D0(B-).

La figura seguente descrive l'assieme di polarizzazione di linea esterna di TWDLCA•40DRF mini-DIN linea RS-485 EIA:



Per eseguire la polarizzazione esterna è possibile procedere in due modi:
- Collegamento esterno dell'assieme di polarizzazione dell'utente tramite un cavo mini-DIN. Per il connettore, vedere la definizione di pin.
- Tramite una spina di polarizzazione configurata per polarizzazione a due fili e un assieme di polarizzazione, tra breve disponibile a catalogo.

Configurazione software

Per configurare il controller in modo che utilizzi una connessione seriale per inviare e ricevere caratteri utilizzando il protocollo Modbus, è necessario:

Passo	Descrizione
1	Configurare la porta seriale per il protocollo Modbus utilizzando TwidoSoft.
2	Creare nell'applicazione una tabella di trasmissione/ricezione utilizzata dall'istruzione EXCHx.

Configurazione della porta

Un controller Twido può utilizzare la propria porta principale 1 o una porta 2 configurata opzionale per il protocollo Modbus. Per configurare una porta seriale per il protocollo Modbus:

Passo	Azione
1	Definire tutti i moduli o le schede di comunicazione supplementari configurati sulla base.
2	Fare clic con il pulsante destro sulla porta e fare clic su Modifica impostazioni di Com. del Controller... e impostare il tipo di porta seriale su "Modbus".
3	Impostare i parametri di comunicazione associati.

Modbus master

La modalità Modbus master permette al controller di inviare una richiesta Modbus a uno slave e di attendere la risposta. La modalità Modbus master è supportata solo tramite l'istruzione EXCHx. La modalità Modbus master supporta Modbus ASCII e RTU.

Le dimensioni massime dei pacchetti dati trasmessi o ricevuti sono di 250 byte. Inoltre, la tabella delle parole associata all'istruzione EXCHx è composta dalle tabelle di controllo, di trasmissione e ricezione.

	Byte più significativo	**Byte meno significativo**
Tabella di controllo	Comando	Lunghezza (Trasmissione/Ricezione)
	Offset ricezione	Offset trasmissione
Tabella di trasmissione	Trasmesso byte 1	Trasmesso byte 2
	...	...
	...	Emesso byte n
	Emesso byte n+1	
Tabella di ricezione	Ricevuto byte 1	Ricevuto byte 2
	...	...
	...	Ricevuto byte p
	Ricevuto byte p+1	

Nota: oltre alle query inviate a singoli slave, il controller master Modbus può iniziare una query **broadcast** a tutti gli slave. In caso di broadcast query, il byte di **comando** deve essere impostato a 00, mentre l'**indirizzo slave** deve essere impostato a 0.

Tabella di controllo

Il byte **Lunghezza** contiene la lunghezza della tabella di trasmissione (al massimo 250 byte), che viene sovrascritta dal numero di caratteri ricevuti alla fine della ricezione, se questa è richiesta.
Questo parametro è la lunghezza in byte della tabella di trasmissione. Se il parametro Offset di trasmissione è uguale a 0, questo parametro sarà uguale alla lunghezza del pacchetto dati di trasmissione. Se il parametro Offset trasmissione non è uguale a 0, un byte della tabella di trasmissione (indicato dal valore di offset) non viene trasmesso e questo parametro è uguale alla lunghezza del pacchetto stesso più 1.
Il byte **Comando** in caso di richiesta Modbus RTU (tranne che per una trasmissione broadcast) deve sempre essere uguale a 1 (trasmissione e ricezione).
Il byte **Offset trasmissione** contiene la posizione (1 per il primo byte, 2 per il secondo byte e così via) nella tabella di trasmissione del byte da ignorare durante la trasmissione dei byte, e viene utilizzato per gestire le trasmissioni associate ai valori di byte/parola nel protocollo Modbus. Ad esempio, se questo byte ne contiene 3, il terzo byte sarà ignorato, così il quarto byte della tabella sarà il terzo ad essere trasmesso.
Il byte **Offset ricezione** contiene la posizione (1 per il primo byte, 2 per il secondo byte e così via) nella tabella di ricezione da aggiungere durante la trasmissione dei byte, e viene utilizzato per gestire le trasmissioni associate ai valori di byte/parola nel protocollo Modbus. Ad esempio, se il byte è uguale a 3, il terzo byte nella tabella viene sostituito da uno ZERO e il terzo byte effettivamente ricevuto viene immesso nella quarta posizione nella tabella.

Tabelle di trasmissione/ ricezione

Quando si utilizza una delle due modalità (Modbus ASCII o Modbus RTU), la tabella di trasmissione viene completata con il contenuto della richiesta prima dell'esecuzione dell'istruzione EXCHx. Al momento dell'esecuzione, il controller determina di quale tipo di connessione si tratta ed esegue tutte le conversioni necessarie ad elaborare la trasmissione e la risposta. I caratteri di inizio, fine e di controllo non sono memorizzati nelle tabelle di trasmissione/ricezione.
Una volta che sono stati trasmessi tutti i byte, il controller passa alla modalità di ricezione e attende di ricevere i byte.
La ricezione può terminare in uno dei seguenti modi:

- rilevazione di un timeout su un carattere o un pacchetto di dati,
- ricezione del carattere di chiusura del pacchetto dati in modalità ASCII,
- completamento della tabella di ricezione.

Le voci **Byte trasmessi X** contengono i dati del protocollo Modbus (codifica RTU) che devono essere trasmessi. Se la porta di comunicazione è configurata per il protocollo Modbus ASCII, alla trasmissione vengono aggiunti i caratteri del pacchetto dati corretti. Il primo byte contiene l'indirizzo della periferica (specifica o generale), il secondo byte contiene il codice della funzione e i restanti contengono le informazioni associate al codice della funzione.

Nota: questa è un'applicazione tipica, ma non definisce tutte le possibilità. Non viene eseguita alcuna convalida dei dati trasmessi.

Le voci **Ricevuto byte X** contengono i dati del protocollo Modbus (codifica RTU) che devono essere ricevuti. Se la porta di comunicazione è configurata per il protocollo Modbus ASCII, dalla risposta vengono rimossi i caratteri del pacchetto dati corretti. Il primo byte contiene l'indirizzo della periferica, il secondo byte contiene il codice della funzione (o codice di risposta) e i restanti contengono le informazioni associate al codice della funzione.

Nota: questa è un'applicazione tipica, ma non definisce tutte le possibilità. Non viene eseguita alcuna convalida dei dati ricevuti, tranne la verifica di checksum.

Modbus slave

La modalità Modbus slave permette al controller di rispondere alle richieste Modbus standard provenienti da un master Modbus.
Se il cavo TSXPCX1031 è collegato al controller, la comunicazione con TwidoSoft viene avviata sulla porta, disabilitando temporaneamente la modalità di comunicazione in esecuzione prima del collegamento del cavo.
Il protocollo Modbus supporta due formati del tipo di connessione: ASCII e RTU. Ciascuno viene definito dall'implementazione del livello fisico, con ASCII che utilizza 7 bit di dati e RTU che utilizza 8 bit di dati.
Quando si utilizza la modalità Modbus ASCII, ciascun byte nel messaggio viene inviato come due caratteri ASCII. Il pacchetto di dati Modbus ASCII inizia con un carattere di inizio (':') e termina con due caratteri di chiusura (CR e LF). Il carattere di fine frame è per impostazione predefinita 0x0A (avanzamento riga) e l'utente può modificare il valore di questo byte durante la configurazione. Il valore di controllo per il pacchetto dati del Modbus ASCII è un semplice complemento di due del pacchetto, escludendo i caratteri di inizio e di fine.
La modalità Modbus RTU non riformatta il messaggio prima della trasmissione; tuttavia, utilizza una diversa modalità di calcolo del checksum, specificata come CRC.
Il tipo di connessione Modbus presenta le seguenti limitazioni:

- Indirizzo 1-247
- Bit: 128 bit su richiesta
- Parole: 125 parole di 16 bit su richiesta

Scambio di messaggi

Per la comunicazione, il linguaggio offre due servizi:

- **Istruzione EXCHx:** per trasmettere/ricevere messaggi.
- **Blocco funzione %MSGx:** per controllare gli scambi di messaggi.

Il controller Twido utilizza il protocollo configurato per la porta specifica durante l'elaborazione di un'istruzione EXCHx.

Nota: ogni porta di comunicazione può essere configurata per protocolli diversi o per lo stesso protocollo. È possibile accedere all'istruzione EXCHx o al blocco funzione %MSGx per ciascuna porta di comunicazione aggiungendo il numero di porta (1 o 2).

Istruzione EXCHx

L'istruzione EXCHx consente a un controller Twido di inviare e/o ricevere informazioni verso/da periferiche Modbus. L'utente definisce una tabella di parole (%MWi:L) contenente le informazioni di controllo oltre ai dati da inviare e/o ricevere (fino a 250 byte in trasmissione e/o ricezione). Il formato della tabella di parole è stato descritto in precedenza.
Uno scambio di messaggi viene eseguito utilizzando l'istruzione EXCHx.

Sintassi: [EXCHx %MWi:L]

dove: x = numero della porta (1 o 2)

L = numero di parole nelle tabelle di parole di comando, trasmissione e ricezione

Il controller Twido deve concludere lo scambio relativo alla prima istruzione EXCHx prima di poter avviare una seconda istruzione di scambio. Nel caso di invio di più messaggi, utilizzare il blocco funzione %MSGx.
L'elaborazione dell'istruzione tramite lista EXCHx avviene immediatamente, con qualsiasi trasmissione avviata dietro il controllo dell'interrupt (anche la ricezione dei dati è controllata dall'interrupt), e questa viene considerata elaborazione in background.

Blocco funzione %MSGx

L'utilizzo del blocco funzione %MSGx è opzionale, può essere impiegato per gestire gli scambi di dati. Il blocco funzione %MSGx ha tre obiettivi:

- **Controllo degli errori di comunicazione**
  La ricerca degli errori consente di verificare che il parametro L (lunghezza della tabella di parole) programmato con l'istruzione EXCHx sia sufficientemente ampio da contenere il messaggio da inviare. Questo viene confrontato con la lunghezza programmata nel byte meno significativo della prima parola della tabella di parole.
- **Coordinamento di più messaggi**
  Per garantire il coordinamento quando si inviano più messaggi, il blocco funzione %MSGx fornisce le informazioni richieste per determinare il momento in cui viene completata la trasmissione del messaggio precedente.
- **Trasmissione di messaggi di priorità**
  Il blocco funzione %MSGx consente di interrompere la trasmissione del messaggio corrente per consentire l'invio immediato di un messaggio urgente.

Il blocco funzione %MSGx dispone di un ingresso e due uscite associati:

I/O	Definizione	Descrizione
R	Azzera ingresso	Impostato a 1: reinizializza la comunicazione o azzera il blocco (%MSGx.E = 0 e %MSGx.D = 1).
%MSGx.D	Comunicazione completata	0: richiesta in corso. 1: comunicazione eseguita se viene ricevuto il segnale di fine trasmissione, carattere finale, errore o azzeramento del blocco.
%MSGx.E	errore	0: lunghezza messaggio corretta e collegamento corretto. 1: se il comando è sbagliato, se la tabella è configurata in modo non corretto, se viene ricevuto un carattere errato (velocità, parità, ecc.) o se la tabella di ricezione è piena.

Limiti

È importante tenere presenti le seguenti limitazioni:

- La presenza e la configurazione della porta 2 (RS232 o RS485) vengono controllate all'accensione o al reset.
- L'elaborazione dei messaggi sulla porta 1 viene terminata al collegamento di TwidoSoft.
- EXCHx o %MSG non possono essere elaborati su una porta configurata come collegamento remoto.
- EXCHx termina l'elaborazione Modbus slave attivo.
- L'elaborazione delle istruzioni EXCHx non viene ritentata in caso di errore.
- L'ingresso Reset (R) può essere utilizzato per terminare l'elaborazione in ricezione di un'istruzione EXCHx.
- Le istruzioni EXCHx possono essere configurate con un timeout per terminare la ricezione.
- Più messaggi sono controllati tramite %MSGx.D.

Errori e condizioni di funzionamento

Se si verifica un errore utilizzando un'istruzione EXCHx, i bit %MSGx.D e %MSGx.E vengono impostati a 1, la parola di sistema %SW63 contiene il codice di errore della porta 1 e %SW64 contiene il codice di errore della porta 2.

Parole di sistema	Utilizzo
%SW63	Codice di errore EXCH1: 0 – Operazione riuscita 1 - Numero di byte da emettere troppo grande (> 250) 2 - Tabella di emissione troppo piccola 3 - Tabella di parole troppo piccola 4 - Overflow tabella di ricezione 5 - Timeout scaduto 6 - Invio 7 - Comando errato nella tabella 8 - Porta selezionata non configurata/disponibile 9 – Errore di ricezione 10 - Impossibile utilizzare %KW se in ricezione 11 - Offset di trasmissione più grande della tabella di trasmissione 12 - Offset di ricezione più grande della tabella di ricezione 13 - Interruzione dell'elaborazione EXCH da controller
%SW64	Codice d'errore EXCH2: Vedere %SW63.

Riavvio del controller master

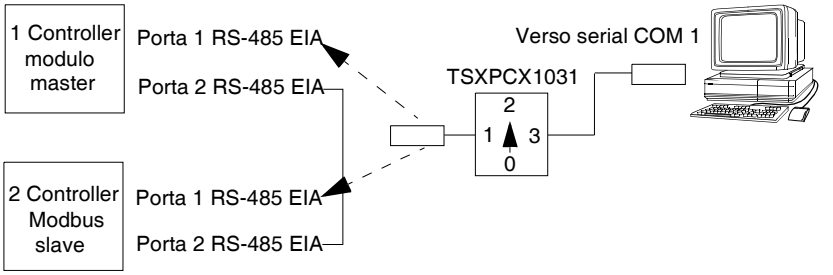
Se un controller master/slave viene riavviato, si verifica uno dei seguenti eventi:

- Un avvio a freddo (%S0 = 1) forza la reinizializzazione delle comunicazioni.
- Un avvio a caldo (%S1 = 1) forza la reinizializzazione delle comunicazioni.
- In modalità Stop, il controller interrompe tutte le comunicazioni Modbus.

Collegamento Modbus - Esempio 1

Per configurare un collegamento Modbus è necessario:
1. Configurare l'hardware.
2. Collegare il cavo di comunicazione Modbus.
3. Configurare la porta.
4. Scrivere un'applicazione.
5. Inizializzare l'editor delle tabelle di animazione.
I diagrammi che seguono illustrano l'uso della richiesta Modbus 3 per leggere le parole d'uscita di uno slave. L'esempio utilizza due controller Twido.

Fase 1: Configurazione hardware:



La configurazione hardware è costituita da due controller Twido. Uno sarà configurato come master Modbus e l'altro come slave Modbus.

Nota: In questo esempio, ciascun controller è configurato per utilizzare EIA RS-485 sulla porta 1 e una porta 2 EIA RS-485 opzionale. Su un controller modulare, la porta opzionale 2 può essere una porta TWDNOZ485D o TWDNOZ485T oppure, se si utilizza TWDXCPODM, può essere TWDNAC485D o TWDNAC485T. Su un controller compatto, la porta opzionale 2 può essere una porta TWDNAC485D o TWDNAC485T.

Per configurare ciascun controller, collegare il cavo TSXPCX1031 alla porta 1 del controller.

Nota: il cavo TSXPCX1031 può essere collegato a un solo controller alla volta, solo sulla porta 1 EIA RS-485.

Quindi collegare il cavo alla porta COM 1 sul PC. Verificare che l'interruttore sia in posizione 2. Scaricare e controllare l'applicazione. Ripetere la procedura per il secondo controller.

Fase 2: Collegare il cavo di comunicazione Modbus:

Connessione mini DIN



Connessione morsettiera



Il cablaggio di questo esempio è una semplice connessione punto a punto. I tre segnali D1(A+), D0(B-) e COM(0V) sono cablati in base allo schema.
Se si utilizza la porta 1 del controller Twido, il segnale DPT (pin 5) deve essere collegato al circuito comune (pin 7). Questa condizione di DPT determina se TwidoSoft è collegato. Se è collegato a terra, il controller utilizzerà la configurazione della porta impostata nell'applicazione per stabilire il tipo di comunicazione.

Fase 3: Configurazione della porta:

Hardware -> Aggiungi opzione TWDNOZ485-	
Hardware => Imposta comunic. controller	
Porta seriale 2	
Protocollo	Modbus
Indirizzo	1
Velocità di trasmissione	19200
Bit di dati	8 (RTU)
Parità	Nessuna
Bit di stop	1
Timeout risposta (x100ms)	10
Ritardo tra frame (ms)	10

Hardware -> Aggiungi opzione TWDNOZ485-	
Hardware => Imposta comunic. controller	
Porta seriale 2	
Protocollo	Modbus
Indirizzo	2
Velocità di trasmissione	19200
Bit di dati	8 (RTU)
Parità	Nessuna
Bit di stop	1
Timeout risposta (x100ms)	100
Ritardo tra frame (ms)	10

In entrambe le applicazioni master e slave, sono configurate le porte opzionali EIA RS-485. Verificare che i parametri di comunicazione del controller siano stati impostati sul protocollo Modbus e su indirizzi diversi.
In questo esempio, il master viene impostato su un indirizzo 1 e lo slave su 2. Il numero di bit è impostato a 8, a indicare che si utilizzerà la modalità Modbus RTU. Se il numero fosse stato 7, si sarebbe utilizzata la modalità Modbus ASCII. L'unico altro valore di default modificato è stato l'incremento del timeout della risposta a 1 secondo.

Nota: dal momento che è stato selezionata la modalità Modbus RTU, il parametro "Fine pacchetto" è stato ignorato.

Fase 4: Scrittura dell'applicazione:

```
LD 1
[%MW0 := 16#0106 ]
[%MW1 := 16#0300 ]
[%MW2 := 16#0203 ]
[%MW3 := 16#0000 ]
[%MW4 := 16#0004 ]
LD 1
AND %MSG2.D
[EXCH2 %MW0:11]
LD %MSG2.E
ST %Q0.0
END
```

```
LD 1
[%MW0 := 16#6566 ]
[%MW1 := 16#6768 ]
[%MW2 := 16#6970 ]
[%MW3 := 16#7172 ]
END
```

Utilizzando TwidoSoft, viene scritto un programma applicativo per il master e per lo slave. Per lo slave vengono semplicemente scritte alcune parole di memoria su una serie di valori noti. Nel master, la tabella delle parole dell'istruzione EXCHx viene inizializzata per leggere 4 parole dallo slave all'indirizzo Modbus 2 a partire dalla posizione %MW0.

Nota: si noti l'uso dell'offset RX impostato in %MW1 del master Modbus. L'offset di tre aggiungerà un byte (valore = 0) alla terza posizione nell'area di ricezione della tabella. Questo allinea le parole nel master in modo che rientrino correttamente nei margini della parola. Senza l'offset, ogni parola di dati sarebbe divisa in due parole nel blocco di scambio. Questo offset viene utilizzato per praticità.

Prima di eseguire l'istruzione EXCH2, l'applicazione verifica il bit di comunicazione associato a %MSG2. Al termine, lo stato di errore di %MSG2 viene rilevato e memorizzato sul primo bit di uscita sull'I/O del controller di base. Per aumentare la precisione, è possibile effettuare ulteriori controlli usando %SW64.

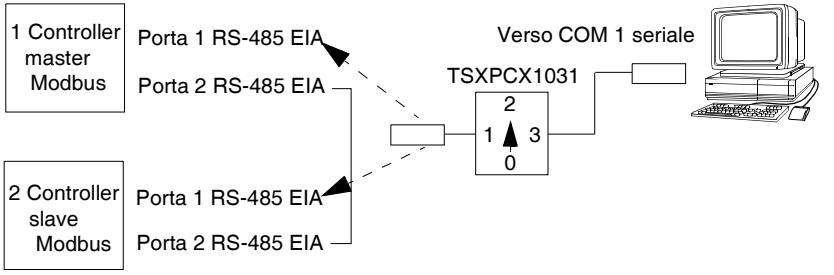
Fase 5: Inizializzazione dell'editor delle tabelle di animazione nel master:

	Indirizzo	Corrente	Ritenuto	Formato
1	%MW5	0203	0000	Esadecimale
2	%MW6	0008	0000	Esadecimale
3	%MW7	6566	0000	Esadecimale
4	%MW8	6768	0000	Esadecimale
5	%MW9	6970	0000	Esadecimale
6	%MW10	7172	0000	Esadecimale

Dopo aver scaricato e impostato ciascun controller per il funzionamento, aprire una tabella di animazione sul master. Esaminare la sezione risposte della tabella per controllare che il codice risposta sia 3 e che sia stato letto il numero di byte corretto. Inoltre, in questo esempio si noti che le parole lette dallo slave (a partire da %MW7) sono allineate correttamente entro i margini della parola sul master.

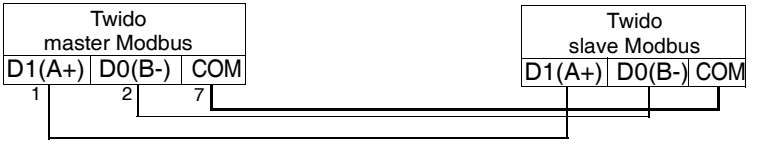
Esempio 2 di collegamento Modbus

I diagrammi che seguono illustrano l'uso della richiesta Modbus 16 per scrivere le parole d'uscita di uno slave. L'esempio utilizza due controller Twido.
Fase 1: Configurazione dell'hardware:



La configurazione hardware è identica a quella dell'esempio precedente.
Fase 2: Connessione del cavo di comunicazione Modbus (RS-485):

Connessione mini DIN



Connessione morsettiera



Il cablaggio delle comunicazioni Modbus è identico a quello dell'esempio precedente.

Fase 3: Configurazione della porta:

Hardware -> Aggiungi opzione TWDNOZ485-

Hardware => Imposta comunic. controller	
Porta seriale 2	
Protocollo	Modbus
Indirizzo	1
Velocità di trasmissione	19200
Bit di dati	8 (RTU)
Parità	Nessuna
Bit di stop	1
Timeout risposta (x100ms)	10
Ritardo tra frame (ms)	10

Hardware -> Aggiungi opzione TWDNOZ485-

Hardware => Imposta comunic. controller	
Porta seriale 2	
Protocollo	Modbus
Indirizzo	2
Velocità di trasmissione	19200
Bit di dati	8 (RTU)
Parità	Nessuna
Bit di stop	1
Timeout risposta (x100ms)	100
Ritardo tra frame (ms)	10

Le configurazioni delle porte sono identiche a quelle dell'esempio precedente.
Fase 4: Scrittura di un'applicazione:

```
LD 1
[%MW0 := 16#010C ]
[%MW1 := 16#0007 ]
[%MW2 := 16#0210 ]
[%MW3 := 16#0010 ]
[%MW4 := 16#0002 ]
[%MW5 := 16#0004 ]
[%MW6 := 16#6566 ]
[%MW7 := 16#6768 ]
LD 1
AND %MSG2.D
[EXCH2 %MW0:11]
LD %MSG2.E
ST %Q0.0
END
```

```
LD 1
[%MW18 := 16#FFFF ]
END
```

Utilizzando TwidoSoft, viene creato un programma applicativo per il master e per lo slave. Per lo slave, scrivere una sola parola di memoria %MW18. In questo modo, viene assegnato spazio sullo slave per gli indirizzi di memoria compresi tra %MW0 e %MW18. Senza l'assegnazione dello spazio, la richiesta Modbus cercherebbe di scrivere in posizioni che non esistono nello slave.
Nel master, la tabella delle parole dell'istruzione EXCH2 viene inizializzata per scrivere 4 parole sullo slave con indirizzo Modbus 2 all'indirizzo %MW16 (10 esadecimale).

Nota: si noti l'uso dell'offset trasmissione impostato in %MW1 dell'applicazione del master Modbus. L'offset di sette permette di eliminare il byte più significativo nella sesta parola (valore 00 esadecimale in %MW5). Ciò consente di allineare i valori dei dati nella tabella di trasmissione della tabella di parole in modo che rientrino correttamente nei limiti delle parole.

Prima di eseguire l'istruzione EXCH2, l'applicazione verifica il bit di comunicazione associato a %MSG2. Al termine, lo stato di errore di %MSG2 viene rilevato e memorizzato sul primo bit di uscita sull'I/O del controller di base. Per aumentare la precisione, è possibile effettuare ulteriori controlli usando %SW64.
Fase 5: Inizializzazione dell'editor delle tabelle di animazione:
Creazione della tabella di animazione seguente sul master:

Indirizzo	Corrente	Ritenuto	Formato
1 %MW0	010C	0000	Esadecimale
2 %MW1	0007	0000	Esadecimale
3 %MW2	0210	0000	Esadecimale
4 %MW3	0010	0000	Esadecimale
5 %MW4	0002	0000	Esadecimale
6 %MW5	0004	0000	Esadecimale
7 %MW6	6566	0000	Esadecimale
8 %MW7	6768	0000	Esadecimale
9 %MW8	0210	0000	Esadecimale
10 %MW9	0010	0000	Esadecimale
11 %MW10	0004	0000	Esadecimale

Creazione della tabella di animazione seguente sullo slave:

Indirizzo	Corrente	Ritenuto	Formato
1 %MW16	6566	0000	Esadecimale
2 %MW17	6768	0000	Esadecimale

Dopo aver scaricato e impostato ciascun controller per l'esecuzione della modalità RUN, aprire una tabella di animazione sul controller slave. I due valori in %MW16 e %MW17 vengono scritti sullo slave. Nel master, la tabella di animazione può essere utilizzata per esaminare la parte di tabella di ricezione dei dati di scambio. Questi dati visualizzano l'indirizzo dello slave, il codice di risposta, la prima parola scritta e il numero di parole scritte a partire da %MW8 nell'esempio precedente.

Richieste standard Modbus

Introduzione

Queste richieste consentono di scambiare parole o bit di memoria tra dispositivi remoti. Il formato della tabella è uguale per entrambe le modalità RTU e ASCII.

Formato	Codice di riferimento
Bit	%Mi
Parola	%MWi

Master Modbus: Lettura di N bit

Questa tabella rappresenta le richieste 01 e 02.

	Tabella Indice	Byte più significativo	Byte meno significativo
Tabella di controllo	0	01 (trasmissione/ricezione)	06 (lunghezza trasmissione) (*)
	1	03 (Offset ricezione)	00 (Offset trasmissione)
Tabella di trasmissione	2	Slave@(1..247)	01 o 02 (Codice di richiesta)
	3	Indirizzo del primo bit da leggere	
	4	N_1 = Numero di bit da leggere	
Tabella di ricezione (dopo la risposta)	5	Slave@(1..247)	01 o 02 (Codice risposta)
	6	00 (byte aggiunto a seguito di un'azione di Offset ricezione)	N_2 = Numero di byte di dati da leggere = [1+(N_1-1)/8], dove [] significa parte integrale
	7	Valore del 1^o byte (valore = 00 o 01)	Valore del 2^o byte (se N_1>1)
	8	Valore del 3^o byte (se N_1>1)	
	...		
	(N_2/2)+6 (se N_2 è pari) (N_2/2+1)+6 (se N_2 è dispari)	Valore di $N_2{}^o$ byte (se N_1>1)	

(*) Questo byte riceve anche la lunghezza della stringa trasmessa dopo la risposta

Master Modbus: Lettura di N parole

Questa tabella rappresenta le richieste 03 e 04.

	Tabella Indice	Byte più significativo	Byte meno significativo
Tabella di controllo	0	01 (trasmissione/ricezione)	06 (lunghezza trasmissione) (*)
	1	03 (Offset ricezione)	00 (Offset trasmissione)
Tabella di trasmissione	2	Slave@(1..247)	03 o 04 (Codice di richiesta)
	3	Indirizzo della prima parola da leggere	
	4	N = Numero di parole da leggere	
Tabella di ricezione (dopo la risposta)	5	Slave@(1..247)	03 o 04 (Codice risposta)
	6	00 (byte aggiunto a seguito di un'azione di Offset ricezione)	2*N (numero di byte letti)
	7	Prima parola letta	
	8	Seconda parola letta (se N>1)	
	...		
	N+6	Ennesima parola letta (se N>2)	

(*) Questo byte riceve anche la lunghezza della stringa trasmessa dopo la risposta

Nota: l'operazione Offset ricezione = 3 aggiunge un byte (valore = 0) alla terza posizione della tabella di ricezione. Ciò consente un buon posizionamento del numero di byte letti e dei valori delle parole lette in questa tabella.

Master Modbus: scrittura di un bit

Questa tabella rappresenta la Richiesta 05.

	Tabella Indice	Byte più significativo	Byte meno significativo
Tabella di controllo	0	01 (trasmissione/ricezione)	06 (lunghezza trasmissione) (*)
	1	00 (Offset ricezione)	00 (Offset trasmissione)
Tabella di trasmissione	2	Slave@(1..247)	05 (Codice di richiesta)
	3	Indirizzo del bit da scrivere	
	4	Valore di bit da scrivere	
Tabella di ricezione (dopo la risposta)	5	Slave@(1..247)	05 (Codice di risposta)
	6	Indirizzo del bit scritto	
	7	Valore scritto	

(*) Questo byte riceve anche la lunghezza della stringa trasmessa dopo la risposta

Nota:
- Questa richiesta non prevede l'uso dell'offset.
- Il frame di risposta è lo stesso di questa richiesta (in un caso normale).
- Per assegnare a un bit il valore 1, la parola associata nella tabella di trasmissione deve contenere il valore FF00H e 0 per assegnare il valore 0.

Master Modbus: scrittura di una parola

Questa tabella rappresenta la Richiesta 06.

	Tabella Indice	Byte più significativo	Byte meno significativo
Tabella di controllo	0	01 (trasmissione/ricezione)	06 (lunghezza trasmissione) (*)
	1	00 (Offset ricezione)	00 (Offset trasmissione)
Tabella di trasmissione	2	Slave@(1..247)	06 (Codice di richiesta)
	3	Indirizzo della parola da scrivere	
	4	Valore della parola da scrivere	
Tabella di ricezione (dopo la risposta)	5	Slave@(1..247)	06 (Codice di risposta)
	6	Indirizzo della parola scritta	
	7	Valore scritto	

(*) Questo byte riceve anche la lunghezza della stringa trasmessa dopo la risposta

Nota:
- Questa richiesta non prevede l'uso dell'offset.
- Il frame di risposta è lo stesso di questa richiesta (in un caso normale).

Master Modbus: scrittura di N bit

Questa tabella rappresenta la Richiesta 15.

	Tabella Indice	Byte più significativo	Byte meno significativo
Tabella di controllo	0	01 (trasmissione/ricezione)	8 + numero di byte (trasmissione)
	1	00 (Offset ricezione)	07 (Offset trasmissione)
Tabella di trasmissione	2	Slave@(1..247)	15 (Codice di richiesta)
	3	Numero del primo bit da scrivere	
	4	N_1 = Numero di bit da scrivere	
	5	00 (byte non inviato, effetto offset)	N_2 = Numero di byte di dati da scrivere = [1+(N_1-1)/8], dove [] significa parte integrale
	6	Valore del 1^o byte	Valore del 2^o byte
	7	Valore del 3^o byte	Valore del 4^o byte
	...		
	(N_2/2)+5 (se N_2 è pari) (N_2/2+1)+5 (se N_2 è dispari)	Valore di $N_2{}^o$ byte	
Tabella di ricezione (dopo la risposta)		Slave@(1..247)	15 (Codice di risposta)
		Indirizzo del 1^o bit scritto	
		Indirizzo dei bit scritti (= N_1)	

Nota:

- L'operazione Offset trasmissione = 7 elimina il 7° byte del frame trasmesso. Consente, inoltre, di assicurare una buona corrispondenza tra i valori delle parole nella tabella di trasmissione.

Master Modbus: scrittura di N parole

Questa tabella rappresenta la Richiesta 16.

	Tabella Indice	Byte più significativo	Byte meno significativo
Tabella di controllo	0	01 (trasmissione/ricezione)	8 + (2*N) (lunghezza trasmissione)
	1	00 (Offset ricezione)	07 (Offset trasmissione)
Tabella di trasmissione	2	Slave@(1..247)	16 (Codice di richiesta)
	3	Indirizzo della prima parola da scrivere	
	4	N = Numero di parole da scrivere	
	5	00 (byte non inviato, effetto offset)	2*N = N. di byte da scrivere
	6	Primo valore di parola da scrivere	
	7	Secondo valore da scrivere	
	...		
	N+5	N valori da scrivere	
Tabella di ricezione (dopo la risposta)	N+6	Slave@(1..247)	16 (Codice di risposta)
	N+7	Indirizzo della prima parola da scrivere	
	N+8	Indirizzo delle parole scritte (= N)	

Nota: l'operazione Offset trasmissione =7 eliminerà il 5° byte MMSB nel frame inviato. Consente, inoltre, di assicurare una buona corrispondenza tra i valori delle parole nella tabella di trasmissione.

Classe di implementazione Transparent Ready (Twido Serial A05, Ethernet A15)

Panoramica

I seguenti codici funzione Modbus sono supportati da Modbus sia seriale sia TCP/IP. Per informazioni dettagliate sul protocollo Modbus, consultare il documento *Modbus Application Protocol (Protocollo per applicazioni Modbus)* disponibile all'indirizzo http://www.modbus-ida.org

Codici funzione Modbus supportati da Twido (MB FC)

La tabella seguente descrive i codici funzione supportati da Twido seriale sia da Modbus TCP/IP:

Codici funzione supportati da MB	Codice sottofunzione supportato	Funzione
1	—	Lettura di bit interni multipli %M
2	—	Lettura di bit interni multipli %M
3	—	Lettura di registri interni multipli %MW
4	—	Lettura di registri interni multipli %MW
5	—	Forzatura di bit interno singolo %M
6	—	Scrittura di registro interno singolo %MW
8	solo 00	Diagnostica eco
15	—	Scrittura di bit interni multipli %M
16	—	Scrittura di registri interni multipli %MW
23	—	Lettura/scrittura di registri interni multipli %MW
43	14	Lettura identificazione dispositivo (funzionalità regolare)

Panoramica sulla comunicazione Ethernet TCP/IP

Funzioni Ethernet

Le informazioni seguenti descrivono le funzioni Ethernet del controller di base TWDLCAE40DRF.
Il controller di base TWDLCAE40DRF è un dispositivo che supporta Ethernet e che implementa il protocollo MBAP (Modbus Application Protocol) tramite TCP/IP. Modbus TCP/IP garantisce la comunicazione peer-to-peer attraverso la rete in una topologia client/server.

Formato di frame

Il controller compatto Twido TWDLCAE40DRF supporta solamente il formato di frame Ethernet II e non i frame IEEE802.3. Si osservi che altri controller di Schneider Electric, come ad esempio le serie Premium e Quantum, supportano i formati di frame sia Ethernet II che IEEE802.3 e sono selezionabili in base al formato di frame. Pertanto, se si pianifica di combinare il controller Twido con controller Premium o Quantum, occorre configurare tutti i controller per l'uso del formato di frame Ethernet II per garantirne la massima compatibilità.

Connessioni TCP

Il controller compatto TWDLCAE40DRF è un dispositivo a 4 canali simultanei in grado di comunicare su una rete Ethernet 100Base-TX. Implementa la negoziazione automatica 100Base-TX e può funzionare anche su una rete 10Base-T. Supporta inoltre una connessione con IP contrassegnato, come configurato nel programma applicativo TwidoSoft (per ulteriori informazioni sull'IP contrassegnato, vedere *Scheda IP contrassegnato, p. 175*).
Il numero massimo di transazioni server supportato dal controller Twido è 1 per connessione TCP.

Indirizzo IP

Il controller compatto TWDLCAE40DRF compie BootP per ottenere un indirizzo di IP da un server di BootP. Per più flessibilità, lei ha tuttavia la capacità per specificare un indirizzo di IP statico attraverso TwidoSoft programma il software, oltre che di definire gli indirizzi IP di subnet mask e gateway.
Inoltre, se il controller compatto TWDLCAE40DRF non ottiene un indirizzo di IP valido dal server di BootP (o se rivela un indirizzo di IP di duplicato quando lei assegna un indirizzo di IP statico), il controllore va nel modo di ripiego ed usa l'indirizzo di è assegnato un indirizzo fisico univoco MAC (indirizzo globale IEEE) memorizzato in modo permanente nel controller compatto. L'indirizzo IP predefinito viene ricavato dall'indirizzo fisico univoco MAC.

Nota: Quando l'indirizzo di IP predefinito è usato, il servizio di Bootp-Cliente è disattivato.

Client/Server Modbus TCP

Un controller TWDLCAE40DRF può essere sia Client sia Server Modbus TCP/IP a seconda che stia interrogando o rispondendo a un dispositivo remoto. Il servizio di messaggeria TCP è implementato tramite la porta TCP 502.

- Il server Modbus implementa la messaggeria Schneider Transparent Ready - standard TR A15.
- Il client Modbus è implementato tramite l'istruzione EXCH3 e la funzione %MSG3. È possibile programmare più istruzioni EXCH3 ma può essere attiva una sola istruzione EXCH3 alla volta. La connessione TCP viene negoziata automaticamente dal controller compatto non appena
  Il client Modbus implementa la messaggeria Schneider Transparent Ready - standard TR A10.

Guida alla configurazione TCP/IP rapida per la comunicazione Ethernet tra PC e controller

Ambito

La presente Guida alla configurazione TCP/IP rapida fornisce informazioni sulla connettività Ethernet e sulla configurazione TCP/IP per permettere all'utente di impostare rapidamente la comunicazione tra il PC che esegue l'applicazione TwidoSoft e il controller Twido tramite una rete Ethernet stand-alone.

Verifica delle impostazioni IP correnti del PC

La procedura seguente descrive come verificare le impostazioni IP correnti del PC. Questa procedura è valida per tutte le versioni del sistema operativo Windows:

Passo	Azione
1	Selezionare **Esegui** dal menu **Start** di Windows.
2	Digitare **"command"** nella casella di testo **Apri** della finestra di dialogo Esegui. **Risultato:** viene visualizzato il prompt **C:\WINDOWS\system32\command.com**.
3	Digitare **"ipconfig"** al prompt di comando.
4	Viene visualizzata la **configurazione IP** di Windows, che mostra i seguenti parametri: Indirizzo IP.....: Subnet Mask.....: Gateway predefinito.....: **Nota:** queste impostazioni IP non possono essere modificate direttamente al prompt di comando. Sono disponibili solo per la consultazione. Se si intende modificare la configurazione IP del PC, consultare la sezione seguente.

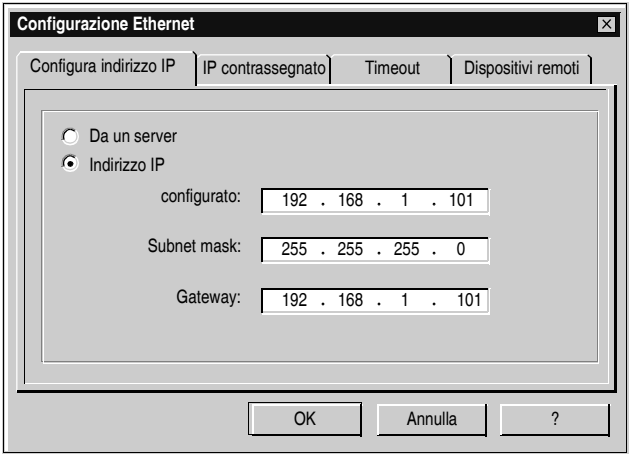
Configurazione delle impostazioni TCP/IP del PC

Le informazioni seguenti permettono di configurare le impostazioni TCP/IP del PC che esegue l'applicazione TwidoSoft per scopi di programmazione e di controllo del controller Twido in rete. La procedura descritta di seguito può essere eseguita su un PC dotato di sistema operativo Windows XP e viene riportata solo a titolo esemplificativo. Per altri sistemi operativi, fare riferimento alle istruzioni per la configurazione TCP/IP fornite nella guida utente del sistema operativo installato sul PC.

Passo	Azione
Nota: se il PC è già installato e la scheda Ethernet è configurata tramite la rete stand-alone esistente, non è necessario modificare le impostazioni dell'indirizzo IP (ignorare i passi da 1 a 6 e passare alla sezione seguente). Eseguire i passi da 1 a 6 di questa procedura solo se si desidera modificare le impostazioni TCP/IP del PC.	
1	Selezionare **Pannello di controllo > Rete** dal menu **Start** di Windows.
2	Fare clic con il pulsante destro del mouse su **Connessione alla rete locale**, la rete stand-alone sulla quale si pianifica di installare il controller Twido, e selezionare **Proprietà**.
3	Selezionare **TCP/IP** dall'elenco di componenti di rete installati e fare clic su **Proprietà**. **Nota:** se il protocollo TCP/IP non figura nell'elenco dei componenti installati, consultare il manuale utente del sistema operativo per informazioni sull'installazione del componente di rete TCP/IP.
4	Viene visualizzata la finestra di dialogo **Proprietà TCP/IP**, che mostra le impostazioni TCP/IP correnti del PC, tra cui **Indirizzo IP** e **Subnet Mask**. **Nota:** su una rete stand-alone, non utilizzare l'opzione **Ottieni automaticamente indirizzo IP**. Il pulsante di opzione **Specifica indirizzo IP** deve essere selezionato e i campi Indirizzo IP e Subnet Mask devono contenere impostazioni IP valide.
5	Immettere un **indirizzo IP statico** valido in formato decimale separato da punti. In una rete stand-alone, si consiglia di specificare un indirizzo IP di rete di classe C (vedere *Indirizzamento IP, p. 165*). Ad esempio, `192.168.1.198` è un indirizzo IP di classe C. **Nota:** l'indirizzo IP che si specifica deve essere compatibile con l'ID della rete esistente. Ad esempio, se la rete esistente supporta indirizzi IP `192.168.1.xxx` (dove `192.168.1` è l'ID di rete e `xxx = 0-255` l'ID dell'host), si può specificare `191.168.1.198` come indirizzo IP valido per il PC (accertarsi che l'ID dell'host `198` sia univoco nella rete).
6	Immettere una **Subnet mask** valida in formato decimale separato da punti. Se la creazione di sottoreti non è usata sulla rete di classe C, si consiglia di specificare una subnet mask predefinita di classe C come ad esempio `255.255.255.0`.

Configurazione delle impostazioni TCP/IP del controller Twido

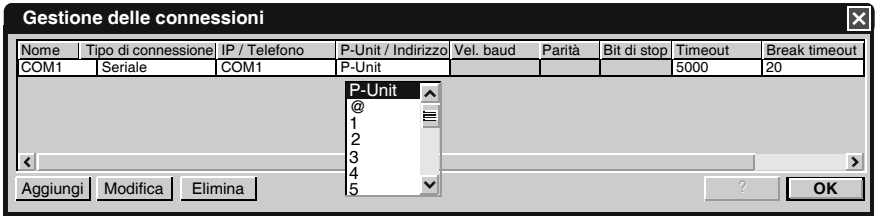
Dopo aver configurato le impostazioni TCP/IP del PC che esegue l'applicazione TwidoSoft, è necessario configurare le impostazioni TCP/P del controller Twido con cui si vuole che TwidoSoft comunichi in rete, come descritto di seguito:

Passo	Azione
1	Collegare un cavo seriale (TSXPCX1031) dal PC che esegue TwidoSoft alla porta RS-485 del controller Twido.
2	Avviare il programma applicativo **TwidoSoft** sul PC.
3	Selezionare un nuovo **Hardware** dal browser dell'applicazione TwidoSoft e scegliere il controller **TWDLCAE40DRF**.
4	Selezionare **Controller > Seleziona una connessione** dalla barra dei menu TwidoSoft e scegliere la porta **COM1**.
5	Fare doppio clic sull'icona **Porta Ethernet** nel browser dell'applicazione TwidoSoft (oppure selezionare **Hardware > Ethernet** dalla barra dei menu) per visualizzare la finestra di dialogo **Configurazione Ethernet**, come mostrato di seguito: 

Passo	Azione
6	Dalla scheda **Configura indirizzo IP**: ● Selezionare il pulsante di opzione **Da un server** per utilizzare il supporto client BootP ed ottenere automaticamente un indirizzo IP dinamico dal server. *(Passare direttamente al punto 10).* **Nota:** il controller TWDLCAE40DRF esegue tre tentativi con un intervallo di 200ms l'uno dall'altro per inviare le richieste BootP al server. Se non viene ricevuta una risposta valida, il controller utilizza l'indirizzo IP predefinito per il posizionamento di sicurezza. ● Selezionare il pulsante di opzione **Configurato** ed iniziare la configurazione dei campi Indirizzo IP statico, Subnet mask e Gateway come spiegato nei passi da 7 a 9. **Nota:** a questo livello si sta impostando solo la configurazione di base della comunicazione tra PC e controller attraverso la rete Ethernet. Non è pertanto ancora necessario configurare le schede IP contrassegnato, Timeout e Dispositivi remoti.
7	Immettere un **Indirizzo IP** statico valido per il controller Twido in formato decimale separato da punti. Questo indirizzo IP deve essere compatibile con l'indirizzo IP del PC configurato nella sezione precedente. **Nota:** gli indirizzi IP del controller Twido e del PC devono condividere lo stesso ID di rete. L'ID host del controller Twido deve essere diverso dall'ID host del PC e univoco nell'ambito della rete. Ad esempio, se l'indirizzo IP di classe C del PC è `192.168.1.198`, un indirizzo valido per il controller Twido è `192.168.1.xxx` (dove`192.168.1` è l'ID di rete e `xxx = 0-197, 199-255` l'ID host).
8	Immettere una **Subnet mask** valida in formato decimale separato da punti. Il controller Twido e il PC che esegue TwidoSoft devono trovarsi sullo stesso segmento di rete. Occorre pertanto immettere una subnet mask identica a quella specificata per il PC. **Nota:** se la creazione di sottoreti non è usata sulla rete di classe C, si consiglia di specificare una subnet mask predefinita di classe C come ad esempio `255.255.255.0`.
9	Immettere un indirizzo di **Gateway** valido in formato decimale separato da punti. **Nota:** se la rete stand-alone non contiene alcun dispositivo gateway, immettere in questo campo l'indirizzo IP del controller Twido impostato nel passo 6.
10	Fare clic su **OK** per salvare le impostazioni di configurazione Ethernet del controller Twido.

Impostazione di una nuova connessione TCP/IP in TwidoSoft

A questo punto si imposta una nuova connessione TCP/IP nell'applicazione TwidoSoft. La nuova connessione TCP/IP dedicata consentirà al PC che esegue TwidoSoft e al controller Twido di comunicare tramite la rete Ethernet.
Selezionare **File** → **Preferenze** dalla barra dei menu di TwidoSoft per richiamare la finestra di dialogo **Gestione delle connessioni**:



Passo	Azione
1	Fare clic sul pulsante **Aggiungi** nella finestra di dialogo **Gestione dei collegamenti**. **Risultato:** viene aggiunta una nuova linea di connessione. La nuova linea presenta le impostazioni di connessioni predefinite, che l'utente dovrà modificare. **Nota:** esistono due possibilità per impostare un nuovo valore in un campo: ● Selezionare il campo desiderato e fare clic sul pulsante **Modifica**. ● Fare doppio clic nel campo desiderato.
2	Nel campo **Nome**, immettere un nome descrittivo per la nuova connessione. Un nome valido può contenere fino a 32 caratteri alfanumerici.
3	Fare clic nel campo **Tipo di connessione** per visualizzare un elenco a discesa che include: TCP/IP, Seriale, Modem (se presente) e USB (se presente). Selezionare **TCP/IP** per impostare una nuova connessione Ethernet tra il PC e un controller Twido connesso a una rete Ethernet.
4	Nel campo **Telefono/IP**, immettere un **Indirizzo IP** valido corrispondente alle informazioni IP relative al controller Twido TWDLCAE40DRF a cui si desidera connettersi. **Indirizzo IP:** Immettere l'indirizzo IP statico specificato per il controller Twido in una delle sezioni precedenti.
5	È possibile completare il campo **Unità P/Indirizzo** quando si seleziona Telefono/IP. Per una connessione TCP/IP, il valore predefinito è **Diretta**. Per una connessione di Tipo seriale, il valore predefinito è **Unità P**. Quando si seleziona una delle suddette opzioni, i successivi tre campi (Velocità, Parità e Bit di stop) sono disabilitati. Se non si conosce l'indirizzo del controller, digitando **@** è possibile selezionarlo successivamente, dopo aver scaricato il programma. *Prima della connessione iniziale, viene visualizzata una finestra a comparsa che consente di selezionare il controller a verso il quale eseguire il trasferimento, all'interno di un intervallo compreso tra 1 e 247 e 1 come valore predefinito dell'indirizzo.*

Passo	Azione
6	Utilizzare le impostazioni predefinite nei campi **Timeout** e **Timeout interruzione** a meno che non si abbiano esigenze specifiche di timeout. Per ulteriori informazioni, vedere *Gestione dei collegamenti Ethernet, p. 182*).
7	Fare clic sul pulsante **OK** per salvare le nuove impostazioni di connessione e chiudere la finestra di dialogo Gestione delle connessioni. **Risultato:** i nomi di tutte le connessione appena aggiunte vengono inseriti nell'elenco a discesa delle connessioni nella finestra di dialogo **File** → **Preferenze**o nel menu **PLC** → **Seleziona una connessione**.

Connessione del controller alla rete

Panoramica

Le informazioni seguenti descrivono come installare il controller compatto TDWLCAE40DRF sulla rete Ethernet.

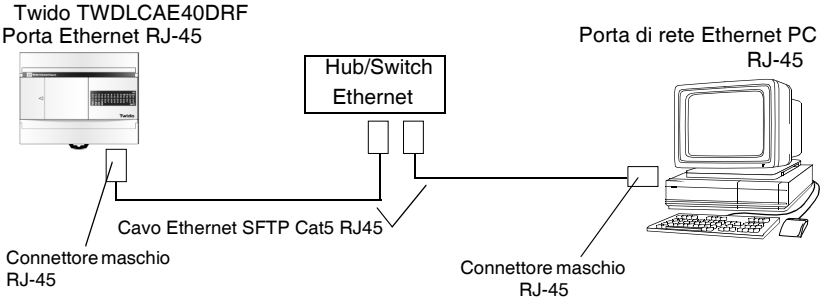
Determinazione del set di indirizzi IP corretto

Consultare l'amministratore di rete per determinare se occorre configurare un nuovo set di indirizzi IP, di gateway e subnet mask. Se l'amministratore assegna nuovi parametri di indirizzi IP, è necessario immettere questi dati manualmente nell'applicazione TwidoSoft. Seguire le istruzioni della sezione *Configurazione TCP/IP, p. 170* riportata più avanti.

Connessione di rete Ethernet

Nota: sebbene la connessione diretta via cavo (tramite un cavo incrociato Ethernet) sia supportata tra il controller Twido TWDLCAE40DRF e il PC che esegue il software di programmazione TwidoSoft, non è un modo di connessione consigliato. Si dovrebbe pertanto preferire una connessione tramite un hub/switch di rete Ethernet.

La figura seguente mostra una connessione di rete Twido tramite un hub/switch di rete Ethernet:



Il controller Twido TWDLCAE40DRF dispone di un connettore RJ-45 per la connessione di rete Ethernet 100BASE-TX con negoziazione automatica. Supporta velocità di rete di 100Mbps e 10 Mbps.

Nota: per la connessione del controller Twido a una rete 100BASE-TX, è necessario usare un cavo Ethernet che sia almeno di categoria 5.

Indirizzamento IP

Panoramica

Questa sezione contiene informazioni sul formato dell'indirizzo IP, sulla subnet mask e sul gateway.

Indirizzo IP

Un indirizzo IP è una quantità a 32 bit espressa in formato decimale separato da punti. Consiste di quattro gruppi di numeri il cui valore è compreso tra 0 e 255, separati l'uno dall'altro da un punto. Ad esempio, 192.168.2.168 è un indirizzo IP in formato decimale separato da punti (si osservi che questo è un indirizzo IP riservato fornito solo a titolo esemplificativo).
Solitamente gli indirizzi IP si dividono in tre categorie, dette classi A, B e C. Le classi possono essere distinte in base al valore del primo numero, come descritto nella tabella seguente:

Primo gruppo decimale	Classe IP
0-127	Classe A
128-191	Classe B
192-223	Classe C

Subnet Mask dell'IP

Un indirizzo IP è costituito da due parti, l'ID di rete e l'ID dell'host. La subnet mask viene usata per dividere la porzione di rete dell'indirizzo IP in modo da creare artificialmente sottoreti con un numero maggiore di ID host. La creazione di sottoreti rappresenta quindi un modo per connettere più unità fisiche a reti logiche. Tutti i dispositivi sulla stessa sottorete condividono lo stesso ID di rete.
Tutti i dispositivi sulla stessa sottorete condividono lo stesso ID di rete.

Nota: se si fa parte di una grande organizzazione, vi sono buone possibilità che la creazione di sottoreti sia implementata dalle reti aziendali. Rivolgersi all'amministratore di rete per ottenere informazioni sulla creazione di sottoreti al momento dell'installazione del nuovo controller Twido sulla rete esistente.

Indirizzo del gateway

Il gateway è il dispositivo di rete detto anche router che fornisce al segmento di rete l'accesso ad altri segmenti di rete sulla rete globale dell'azienda, l'accesso a Internet o a una Intranet remota.
L'indirizzo del gateway usa lo stesso formato decimale separato da punti dell'indirizzo IP descritto sopra.

Nota: rivolgersi all'amministratore di rete per ottenere informazioni sul gateway al momento dell'installazione del nuovo controller Twido sulla rete esistente.

Assegnazione di indirizzi IP

Panoramica

Questa sezione contiene informazioni che permettono di determinare il tipo di indirizzo IP assegnabile al controller Twido TWDLCAE40DRF che si vuole installare sulla rete.

Installazione su una rete stand-alone

Il controller Twido TWDLCAE40DRF è previsto per l'installazione su una rete Ethernet stand-alone.

Nota: una rete viene detta stand-alone quando non è collegata a Internet o a una Intranet aziendale.

Ottenimento di un indirizzo via BootP

Indirizzo BootP Served: Se si sceglie **Da un server** nella scheda **Configura indirizzo IP**, il controller Twido prova prima ad ottenere un indirizzo IP dal server BootP.
Il processo BootP aspetta una risposta dal server BootP. Se non viene ricevuto un indirizzo IP valido dopo la trasmissione della richiesta BootP, Twido assume la configurazione IP predefinita derivata dall'indirizzo MAC (vedere *Indirizzo MAC e indirizzo IP predefinito del Controller, p. 166* più avanti.)

MAC Indirizzo e indirizzo IP predefinito del controller

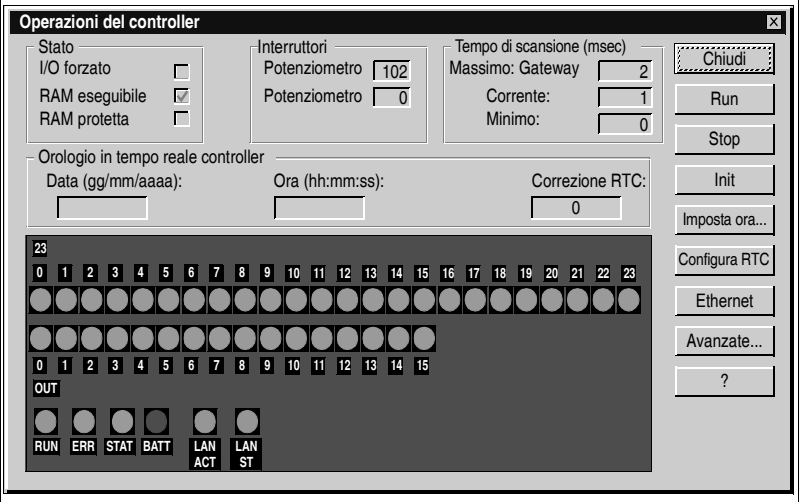
Indirizzo MAC: Ogni controller Twido TWDLCAE40DRF ha il proprio indirizzo MAC impostato in fabbrica che è un indirizzo a 48 bit unico a livello mondiale assegnato a ogni dispositivo Ethernet.
Indirizzo IP predefinito: L'indirizzo IP Ethernet predefinito del controller Twido è derivato dall'indirizzo MAC univoco.
L'indirizzo IP predefinito espresso in formato decimale separato da punti si definisce nel seguente modo:
`085.016.xxx.yyy`, dove:

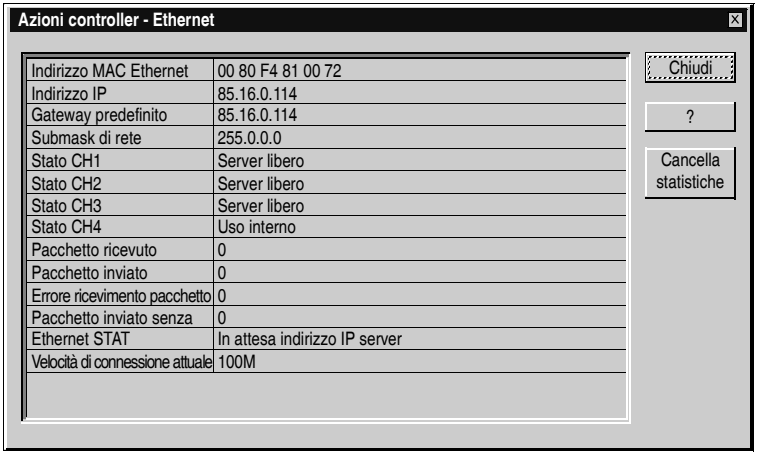
- `085.016.` è un'intestazione fissa condivisa da tutti gli indirizzi IP derivati dall'indirizzo MAC,
- `xxx` e `yyy` sono gli ultimi due numeri dell'indirizzo MAC del dispositivo.

Ad esempio, l'indirizzo IP derivato dall'indirizzo MAC `00.80.F4.81.01.11` è `085.016.001.17`.

Verifica dell'indirizzo MAC e dell'indirizzo IP corrente del controller

Per verificare l'indirizzo MAC e l'indirizzo IP corrente del controller Twido, oltre che le impostazioni di configurazione IP (indirizzi di subnet mask e gateway) e lo stato della connessione Ethernet, procedere nel seguente modo:

Passo	Azione
1	Nel programma applicativo TwidoSoft, selezionare **Controller** dalla barra dei menu.
2	Selezionare **Verifica controller** nell'elenco delle voci del menu. **Risultato:** viene visualizzata la finestra di dialogo **Operazioni del controller**, che mostra i LED Twido su un pannello, come mostrato nella figura seguente: 

Passo	Azione
3	Fare clic sul pulsante **Ethernet** situato sulla destra dello schermo per accedere ai parametri di connessione. **Risultato:** viene visualizzata la tabella **Operazioni di controllo - Ethernet**, che contiene informazioni sugli indirizzi MAC, IP corrente, subnet mask e gateway, oltre che sulla connessione Ethernet, come mostrato nella figura seguente: 
4	Si noti che l'indirizzo **MAC** univoco del controller Twido è riportato nella prima riga della tabella Ethernet.
5	Le informazioni IP contenute nella tabella variano a seconda delle impostazioni dell'utente nella scheda **Configura indirizzo IP** della finestra di dialogo **Configurazione Ethernet** (vedere la scheda *Scheda Configura indirizzo IP, p. 172*): ● se si è selezionato **Dal server** nella scheda Configura indirizzo IP, la tabella presenta l'indirizzo IP predefinito (derivato dall'indirizzo MAC) del controller Twido, oltre che la subnet mask e il gateway predefiniti. Si noti che l'indirizzo IP predefinito è utilizzato solo nella modalità di posizionamento di sicurezza, se non si riesce ad ottenere un indirizzo IP del server BootP valido. Quando si utilizza un canale come UDP per BootP, lo stato del canale indica **Per uso interno.** ● se si è selezionato **Configurato** nella scheda Configura indirizzo IP, la tabella presenta le impostazioni correnti per indirizzo IP, subnet mask e gateway precedentemente immesse nella scheda Configura indirizzo IP. Nota: i campi restanti contengono informazioni sullo stato corrente della connessione Ethernet. Per maggiori informazioni, vedere (Vedere Guida all'uso di TwidoSoft - Guida in linea).

Indirizzi IP privati

Se la network è stand-alone (isolata da Internet), si può assegnare al proprio nodo di rete (controller Twido) un qualsiasi indirizzo IP arbitrario (purché sia conforme alla regola di notazione IANA e non sia in conflitto con l'indirizzo IP di un altro dispositivo già collegato alla rete).
Gli indirizzi IP privati permettono un indirizzamento IP arbitrario su una rete stand-alone. Si noti che gli indirizzi compresi nello spazio di indirizzi privati saranno univoci solo all'interno dell'azienda.
La tabella seguente descrive lo spazio di indirizzi IP privati:

Rete	Intervallo valido per indirizzi IP privati
Classe A	10.0.0.0 -> 10.255.255.255
Classe B	172.16.0.0 -> 172.31.255.255
Classe C	192.168.0.0 -> 192.168.255.255

Assegnazione di un indirizzo IP al controller

Raramente le reti odierne sono del tutto isolate da Internet o dal resto della rete Ethernet aziendale. Pertanto, se si installa il controller di base Twido su una rete esistente, non assegnare un indirizzo IP arbitrario senza aver prima consultato l'amministratore di rete. Per l'assegnazione di un indirizzo IP al controller, attenersi alle istruzioni riportate di seguito.

Nota: è pratica comune usare indirizzi IP di classe C sulle reti stand-alone.

Configurazione TCP/IP

Panoramica

Le istruzioni seguenti spiegano come impostare la configurazione Ethernet TCP/IP del controller compatto Twido TWDLCAE40DRF.

Nota: la configurazione TCP/IP può essere eseguita solo quando l'applicazione TwidoSoft è in modalità non in linea.

⚠ ATTENZIONE

FUNZIONAMENTO NON PREVISTO DI APPARECCHIATURE

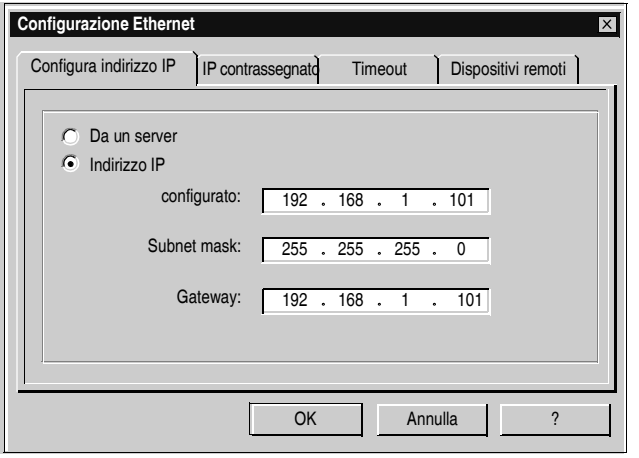
La presenza di due dispositivi con lo stesso indirizzo IP può provocare il funzionamento errato della rete.

- Verificare che al dispositivo sia assegnato un indirizzo IP univoco.
- Richiedere sempre l'indirizzo IP dall'amministratore di rete, per evitare il rischio di duplicati.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

Visualizzazione della finestra di dialogo Configurazione Ethernet

Per visualizzare la finestra di dialogo **Configurazione Ethernet**, procedere come segue:

Passo	Azione
1	Aprire il **Browser dell'applicazione**, come mostrato nella figura seguente. **Risultato:**  **Nota:** accertarsi che come hardware corrente sia selezionato un dispositivo che supporta Ethernet, come ad esempio il controller TWDLCAE40DRF In caso contrario l'opzione hardware Porta Ethernet non viene visualizzata.
2	Fare doppio clic sull'icona **Porta Ethernet** per visualizzare la finestra di dialogo **Configurazione Ethernet**, come mostrato di seguito. **Risultato:**  **Nota:** esistono due modi per visualizzare la schermata Configurazione Ethernet: **1.** Fare clic con il pulsante destro del mouse sull'icona **Porta Ethernet** e selezionare **Modifica** nell'elenco a comparsa. **2.** Selezionare **Hardware > Ethernet** dalla barra dei menu TwidoSoft.

Configurazione TCP/IP

Le sezioni seguenti spiegano come configurare i parametri del controller TCP/IP Twido TWDLCAE40DRF mediante le schede **Configura indirizzo IP, IP contrassegnato, Timeout** e **Dispositivi remoti**.

Scheda Configura indirizzo IP

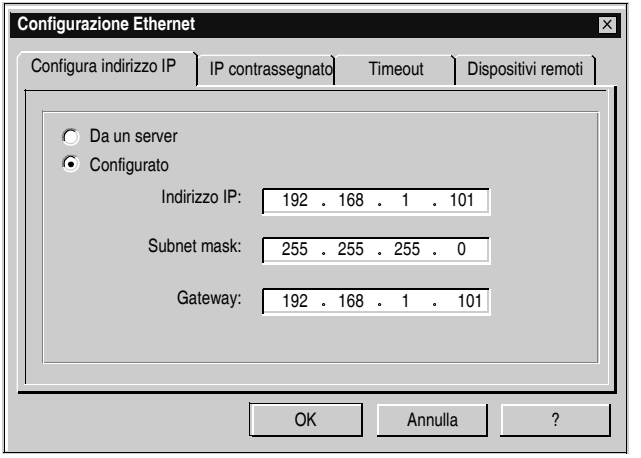
Panoramica

Le informazioni seguenti descrivono come impostare la scheda Configura indirizzo IP della finestra di dialogo Configurazione Ethernet.

Nota: l'indirizzo IP del controller Twido può essere configurato solo quando l'applicazione TwidoSoft è in modalità offline

Scheda Configura indirizzo IP

La figura seguente mostra un esempio di schermata della scheda Configura indirizzo IP con esempi di indirizzi IP, di subnet mask e gateway configurati manualmente dall'utente:



Configurazione della scheda Configura indirizzo IP

Le informazioni seguenti descrivono come configurare i vari campi della scheda Configura indirizzo IP:

Campo	Configurazione
Da un server	Selezionare questo pulsante di opzione se non si vuole impostare manualmente l'indirizzo IP del controller Twido (le caselle di testo Indirizzo IP, Subnet mask e Gateway sono inattive). Il controller Twido (BootP client) utilizzerà l'indirizzo IP assegnato automaticamente dal server. Il controller Twido sceglierà di utilizzare l'indirizzo IP predefinito (dello modalità di posiz. di sicurezza) se non può ottenere un indirizzo IP del server valido dopo aver effettuato tre tentativi intervallati di 200 ms. (Notare che il controller Twido invia periodicamente delle richieste al server con intervalli di 15s fino a quando non ottiene un indirizzo IP valido). L'indirizzo IP predefinito dell'interfaccia Ethernet è derivato dal suo indirizzo MAC. Si noti che l'indirizzo IP predefinito non verrà cambiato automaticamente quando un qualunque canale (ad eccezione del canale per uso interno) del Controller è attivo. **Nota:** per ulteriori informazioni sugli indirizzi BootP e MAC, vedere *Assegnazione di indirizzi IP, p. 166*.
Configurato	Selezionare questo pulsante di opzione per configurare manualmente gli indirizzi IP, di subnet mask e gateway. **Nota:** per ottenere i parametri IP validi per la propria rete, rivolgersi all'amministratore di rete o di sistema.
Indirizzo IP	Immettere l'indirizzo IP statico in formato decimale separato da punti. **Attenzione:** per garantire una buona comunicazione tra i dispositivi, è necessario che gli indirizzi IP del PC che esegue l'applicazione TwidoSoft e del controller Twido condividano lo stesso ID di rete. **Nota:** per garantire una buona comunicazione di rete, ogni dispositivo collegato deve avere un indirizzo IP univoco. Quando è collegato alla rete, il controller Twido verifica che non esistano indirizzi IP doppi. Se nella rete viene rilevato un indirizzo IP doppio, il LED LAN ST del controller Twido lampeggia per 4 volte di seguito. In questo campo occorre quindi immettere un nuovo indirizzo IP non doppio.

Campo	Configurazione
Subnet mask	Immettere il valore di subnet mask valido assegnato al controller dall'amministratore di rete. Si noti che questo campo non può rimanere vuoto; è quindi necessario immettere un valore. Per impostazione predefinita, l'applicazione TwidoSoft calcola automaticamente e visualizza un valore di subnet mask basato sull'IP di classe fornito nel campo precedente Indirizzo IP. I valori predefiniti di subnet mask, a seconda della categoria dell'indirizzo IP di rete Twido, seguono questa regola: Rete classe A -> Valore di subnet mask predefinito: 255.0.0.0 Rete classe B -> Valore di subnet mask predefinito: 255.255.0.0 Rete classe C -> Valore di subnet mask predefinito: 255.255.255.0 **Attenzione:** per garantire una buona comunicazione tra le apparecchiature, il valore di subnet mask configurato sul PC che esegue l'applicazione TwidoSoft deve coincidere con quello del controller Twido. **Nota:** salvo requisiti speciali del controller Twido, usare il valore di subnet mask predefinito.
Gateway	Immettere l'indirizzo IP del gateway. Nella LAN il gateway deve trovarsi sullo stesso segmento del controller Twido. Questa informazione viene generalmente fornita dall'amministratore di rete. Si noti che l'applicazione non fornisce alcun valore predefinito; occorre pertanto immettere un indirizzo di gateway valido in questo campo. **Nota:** se la rete non comprende alcun dispositivo gateway, immettere nel campo Gateway l'indirizzo IP del controller Twido.

Scheda IP contrassegnato

Panoramica

Le informazioni seguenti descrivono come impostare la scheda IP contrassegnato della finestra di dialogo Configurazione Ethernet.

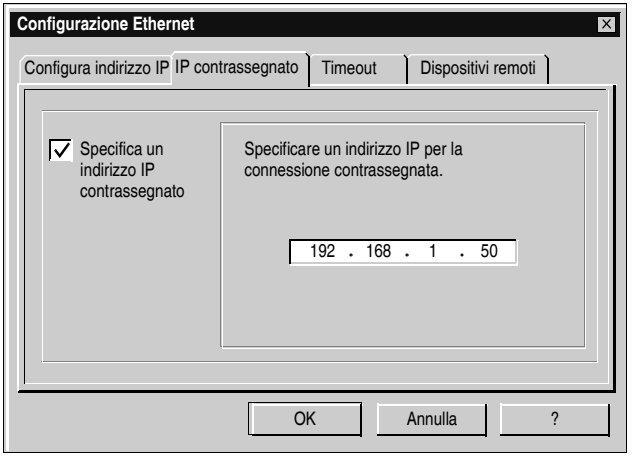
Nota: l'IP contrassegnato (Marked IP) può essere configurato solo quando il programma TwidoSoft si trova in modalità Offline.

Definizione della funzione IP contrassegnato

Questa funzione consente di riservare uno dei quattro canali di connessione Ethernet TCP supportati dal controller Twido per un client particolare designato come IP contrassegnato.
Garantisce inoltre che un canale TCP sia riservato e sempre disponibile per la comunicazione con un dispositivo remoto specificato, anche se il controllo inattività è disattivato (timeout di inattività impostato a "0").

Scheda IP contrassegnato

La figura seguente mostra un esempio di schermata della scheda IP contrassegnato con un esempio di indirizzo IP contrassegnato immesso dall'utente:



Configurazione della scheda IP contrassegnato

Procedere nel seguente modo per configurare la scheda IP contrassegnato:

Passo	Azione
1	Selezionare la casella **Specifica un indirizzo IP contrassegnato** per attivare la funzione IP contrassegnato. Si noti che per impostazione predefinita la funzione IP contrassegnato è disabilitata. **Risultato:** la casella dell'indirizzo IP diventa attiva nella parte destra del frame, come mostrato nella figura precedente.
2	Immettere l'indirizzo IP dell'host client di cui si desidera contrassegnare l'IP nella casella dell'indirizzo IP. **Nota:** questo campo non contiene un valore predefinito. Occorre immettere l'indirizzo IP del dispositivo contrassegnato oppure deselezionare la casella Specifica un indirizzo IP contrassegnato per disattivare questa funzione.

Scheda Timeout

Panoramica

Le informazioni seguenti descrivono come impostare la scheda Timeout contrassegnato della finestra di dialogo Configurazione Ethernet.

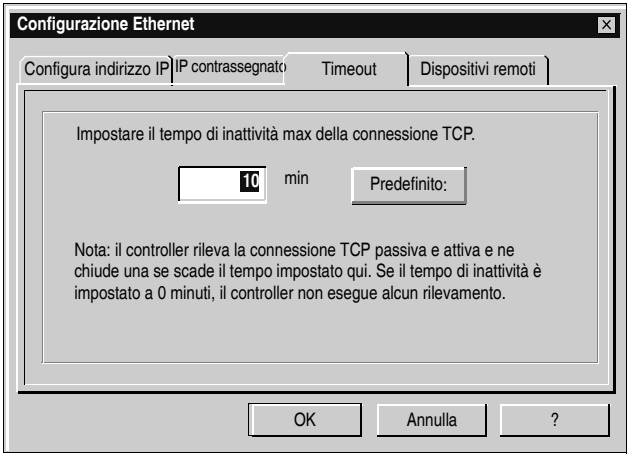
Nota: è possibile configurare il Timeout del controller Twido solo quando il programma applicativo TwidoSoft si trova in modalità non in linea.

Definizione di timeout

Il timeout applica un timeout di inattività a tutte le connessioni Ethernet TCP correnti del controller Twido. Il timeout di inattività è il tempo in cui uno o più dei quattro canali di connessione Ethernet TCP possono rimanere inattivi prima che la connessione host del client remoto con questo canale si interrompa.
Nota: il timer di inattività viene azzerato quando vi è traffico di dati sul canale di connessione monitorato.

Scheda Timeout

La figura seguente mostra un esempio di schermata della scheda Timeout con il valore predefinito di 10 min del timer di inattività:



Configurazione della scheda Timeout

Per impostare il timer inattività, immettere direttamente il tempo trascorso in minuti nella casella di testo **min**, come mostrato nella figura precedente.

Nota:

1. Il tempo trascorso predefinito è 10 minuti. Dopo aver immesso un valore, per **ripristinare** il tempo trascorso configurato a 10 minuti, fare clic sul pulsante **Predefinito**.
2. Per **disabilitare** la funzione di timeout, impostare il tempo trascorso a **0**. In tal modo, il controller Twido non esegue più controlli di inattività. Di conseguenza le connessioni TCP vengono mantenute per un periodo di tempo indefinito.
3. Il tempo di inattività massimo consentito è di 255 minuti.

Scheda Dispositivi remoti

Panoramica

Le informazioni seguenti descrivono come configurare la scheda Dispositivi remoti della finestra di dialogo Configurazione Ethernet nel caso in cui si desideri utilizzare l'istruzione EXCH3 per far funzionare il controller Twido come client Modbus TCP/IP.

Nota: la scheda Dispositivi remoti del controller Twido può essere configurata solo quando l'applicazione TwidoSoft è in modalità offline.

Informazioni preliminari

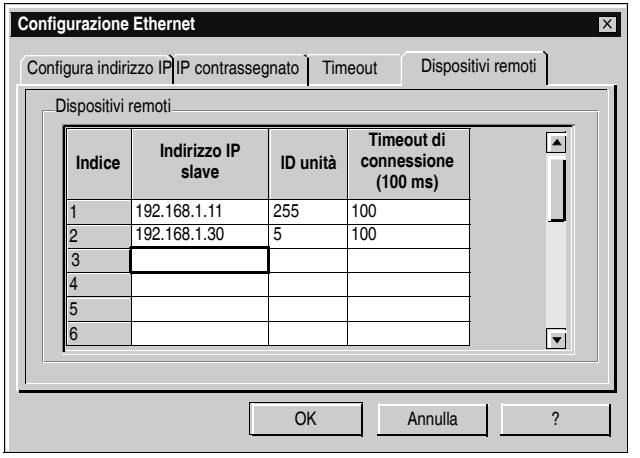
È necessario configurare i dispositivi remoti solo sul controller che deve utilizzare l'istruzione EXCH3 del client Modbus TCP/IP (master legacy Modbus).

Tabella Dispositivi remoti

La tabella Dispositivi remoti contiene informazioni sui controller remoti (che fungono da server Modbus TCP/IP) sulla rete Ethernet che possono essere interrogati dal client Modbus TCP/IP usando l'istruzione EXCH3. Occorre pertanto configurare la tabella Dispositivi remoti in modo che il controller client Modbus TCP/IP possa interrogare i controller server Modbus TCP/IP tramite la rete.

Scheda Dispositivi remoti

La figura seguente mostra un esempio di schermata della scheda Dispositivi remoti configurata sul controller Twido che funge da client Modbus TCP/IP:



Configurazione della scheda Dispositivi remoti

Le informazioni seguenti descrivono come configurare i vari campi della scheda Dispositivi remoti.

Campo	Configurazione
Indice	Questo è un campo di sola lettura che contiene l'indice MBAP associato all'indirizzo IP di rete Ethernet del dispositivo remoto (server Modbus TPC/IP specificato nel campo Indirizzo IP slave). L'indice MBAP è richiamato dall'istruzione EXCH3 come uno degli argomenti di funzione per identificare quale controller remoto specificato nella tabella è interrogato dal client Modbus TCP/IP. **Nota:** in questa tabella è possibile specificare fino a 16 dispositivi remoti diversi indicati con i numeri da 1 a 16.
Indirizzo IP slave	Immettere in questo campo l'indirizzo IP del dispositivo remoto (server Modbus TCP/IP). **Nota:** gli indirizzi IP slave devono essere configurati a partire dall'indice 1, in ordine crescente e in maniera consecutiva. Ad esempio, non è consentito configurare gli indirizzi IP slave dell'indice 1 e poi del 3, in quanto prima occorre configurare l'indice 2.
ID unità	Immettere in questo campo L'ID unità Modbus (o l'indirizzo di protocollo). Un ID unità valido può variare da 0 a 255. L'impostazione predefinita è 255. Un ID unità (diverso da 255) rende possibili le comunicazioni con un indirizzo di dispositivo remoto tramite un ponte o gateway Modbus. Se il dispositivo di destinazione è un altro controller Twido o un dispositivo legacy Modbus installato su un altro indirizzo di collegamento seriale di bus tramite un gateway, è possibile impostare l'ID unità del dispositivo remoto in modo corrispondente. In questo campo occorre impostare l'IP slave come l'indirizzo IP del gateway o del ponte e l'ID unità come l'indirizzo di collegamento seriale Modbus del dispositivo di destinazione.
Timeout di connessione (100 ms)	Specificare il tempo in unità di 100 ms durante il quale il controller Twido continua a tentare di stabilire una connessione TCP con il dispositivo remoto. Se scade il timeout e la connessione non è ancora stabilita, il controller Twido non effettua altri tentativi finché non arriva la successiva richiesta di connessione con l'istruzione EXCH3. Un'impostazione valida per il timeout può essere compresa tra 0 e 65535 (che equivale a 0 - 6553.5 s). L'impostazione predefinita è 100.

Visualizzazione della configurazione Ethernet

Panoramica

L'**Editor di configurazione** di TwidoSoft permette di visualizzare la configurazione Ethernet del controller Twido.

Visualizzazione della configurazione Ethernet

Per visualizzare le impostazioni correnti della configurazione Ethernet tramite l'Editor di configurazione, procedere nel seguente modo.

Passo	Azione
1	Selezionare **Programma > Editor di configurazione** dalla barra dei menu TwidoSoft.
2	Fare clic sul collegamento **ETH** nella barra delle applicazioni dell'Editor di configurazione e fare doppio clic sul collegamento **Porta Ethernet** nel Browser dell'applicazione.
3	I parametri di configurazione Ethernet TCP/IP vengono visualizzati in una tabella simile a quella illustrata di seguito:  Configurazione Ethernet **Configurazione indirizzo IP** Indirizzo IP 192 . 168 . 1 . 101 Subnet mask 255 . 255 . 255 . 0 Indirizzo gateway 192 . 168 . 1 . 101 IP contrassegnato 192 . 168 . 1 . 50 **Server remoto Indirizzo IP slave** / **ID unità** / **Timeout di connessione** 192 . 168 . 1 . 11 — 255 — 100 192 . 168 . 1 . 30 — 5 — 100 192 . 168 . 1 . 50 — 255 — 1500 192 . 168 . 1 . 16 — 255 — 1500 192 . 168 . 1 . 20 — 255 — 100
4	A questo punto, se si sono appena apportate delle modifiche alle impostazioni di configurazione Ethernet TCP/IP del controller Twido, si può decidere se salvare le modifiche o se annullarle e ripristinare la configurazione precedente, come spiegato di seguito: ● Selezionare **Strumenti > Accetta modifiche** nella barra dei menu TwidoSoft per salvare le modifiche apportate alla configurazione Ethernet TCP/IP. ● Selezionare **Strumenti > Annulla modifiche** per annullare le modifiche e ripristinare le impostazioni della precedente configurazione Ethernet TCP/IP. ● Selezionare **Strumenti > Modifica...** per tornare alla finestra di dialogo Configurazione Ethernet e modificare le impostazioni di configurazione TCP/IP. ● Selezionare **PLC > Trasferimento PC=>Controller...** per scaricare il file di configurazione completo nel controller Twido.

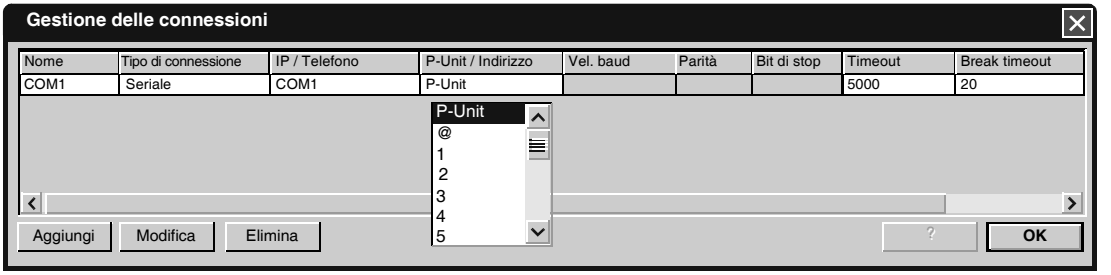
Gestione delle connessioni Ethernet

Panoramica

Le informazioni seguenti descrivono come configurare, aggiungere, eliminare, selezionare una connessione Ethernet TCP/IP da PC a controller.

Impostazione di una nuova connessione TCP/IP

Per impostare una connessione Ethernet TCP/IP tra il PC che esegue l'applicazione TwidoSoft e un controller TWDLCAE40DRF installato sulla rete, procedere come segue.
Selezionare **File** → **Preferenze** dalla barra dei menu di TwidoSoft per richiamare la finestra di dialogo **Gestione delle connessioni**:



Passo	Azione
1	Fare clic sul pulsante **Aggiungi** nella finestra di dialogo **Gestione dei collegamenti**. **Risultato:** viene aggiunta una nuova linea di connessione. La nuova linea presenta le impostazioni di connessioni predefinite, che l'utente dovrà modificare. **Nota:** esistono due possibilità per impostare un nuovo valore in un campo: ● Selezionare il campo desiderato e fare clic sul pulsante **Modifica**. ● Fare doppio clic nel campo desiderato.
2	Nel campo **Nome**, immettere un nome descrittivo per la nuova connessione. Un nome valido può contenere fino a 32 caratteri alfanumerici.
3	Fare clic nel campo **Tipo di connessione** per visualizzare un elenco a discesa che include: TCP/IP, Seriale, Modem (se presente) e USB (se presente). Selezionare **TCP/IP** per impostare una nuova connessione Ethernet tra il PC e un controller Twido connesso a una rete Ethernet.
4	Nel campo **Telefono/IP**, immettere un **Indirizzo IP** valido corrispondente alle informazioni IP relative al controller Twido TWDLCAE40DRF a cui si desidera connettersi. **Indirizzo IP:** Immettere l'indirizzo IP statico specificato per il controller Twido in una delle sezioni precedenti.

Passo	Azione
5	È possibile completare il campo **Unità P/Indirizzo** quando si seleziona Telefono/IP. Per una connessione TCP/IP, il valore predefinito è **Diretta**. Per una connessione di Tipo seriale, il valore predefinito è **Unità P**. Quando si seleziona una delle suddette opzioni, i successivi tre campi (Velocità, Parità e Bit di stop) sono disabilitati. Se non si conosce l'indirizzo del controller, digitando **@** è possibile selezionarlo successivamente, dopo aver scaricato il programma. Prima della connessione iniziale, viene visualizzata una finestra a comparsa che consente di selezionare il controller verso il quale eseguire il trasferimento, all'interno di un intervallo compreso tra 1 e 247 e 1 come valore predefinito dell'indirizzo.
6	Nel campo **Timeout**, immettere un valore di timeout in millisecondi (ms) per stabilire una connessione con il controller Twido. Se scade il timeout e il PC non è ancora riuscito a stabilire una connessione con il controller, l'applicazione TwidoSoft rinuncia a stabilire il collegamento. Per eseguire un nuovo tentativo di connessione, selezionare **Controller** → **Seleziona connessione** dalla barra dei menu di TwidoSoft. **Nota:** il valore predefinito di timeout è **500** ms. Il valore massimo è 65535 x 100 ms (6553,5 s).
7	Il valore **Timeout interruzione** è il tempo massimo che intercorre tra una query TCP/IP Modbus e il ricevimento del frame di risposta. Se questo valore viene superato senza che si riceva il frame di risposta richiesto, l'applicazione TwidoSoft interrompe la connessione tra PC e controller. **Nota:** il valore predefinito di Timeout interruzione è **20** ms. Impostare un valore diverso da zero.
8	Fare clic su **OK** per salvare le nuove impostazioni di connessione e chiudere la finestra di dialogo Gestione delle connessioni. **Risultato:** i nomi di tutte le connessione appena aggiunte vengono inseriti nell'elenco a discesa delle connessioni nella finestra di dialogo **File** → **Preferenze** oppure nel menu **Controller** → **Seleziona connessione**.

Modifica ed eliminazione di una connessione TCP/IP

Per eliminare le connessioni Ethernet TCP/IP esistenti o modificarne i parametri, procedere come segue:

- Per eliminare una connessione dalla finestra di dialogo di gestione Ethernet, selezionare il nome di una connessione e fare clic sul pulsante **Elimina**. Si noti che dopo l'eliminazione tutti i parametri di connessione vengono irrimediabilmente perduti.
- Per modificare i parametri di una connessione esistente, selezionare il campo desiderato e fare clic sul pulsante **Modifica**. A questo punto, è possibile immettere un nuovo valore nel campo selezionato.

Indicatori LED Ethernet

Panoramica

Due indicatori LED relativi alle comunicazioni Ethernet sono situati sul pannello del LED, sul pannello frontale del controller TWDLCAE40DRF e sul pannello software accessibile tramite il percorso **Controller > Verifica controller** nell'applicazione TwidoSoft. Questi LED sono contrassegnati nel seguente modo:

- LAN ACT
- LAN ST

I LED Ethernet permettono di monitorare continuamente lo stato e la diagnostica delle connessioni tramite porta Ethernet.

Stato dei LED

La tabella seguente descrive lo stato di entrambi gli indicatori LED Ethernet **LAN ACT** e **LAN ST** .

LED	Stato	Colore	Descrizione
LAN ACT	Spento	-	Nessun segnale Ethernet sulla porta RJ-45.
	Fisso	Verde	Segnale a battimento per collegamento 10BASE-TX che indica una connessione a 10 Mbps.
	Lampeggiante		Pacchetti di dati inviati o ricevuti tramite connessione 10BASE-TX.
	Fisso	Giallo	Segnale a battimento per collegamento 100BASE-TX che indica una connessione a 100 Mbps.
	Lampeggiante		Pacchetti di dati inviati o ricevuti tramite connessione 100BASE-TX.
LAN ST	Fisso	Verde	Controller di base acceso. Porta Ethernet pronta a comunicare tramite la rete.
	Lampeggio rapido		Inizializzazione Ethernet all'accensione.
	2 lampeggi, lungo spegnimento		Nessun indirizzo MAC valido.
	3 lampeggi, lungo spegnimento		Tre cause possibili: ● Nessun battimento di collegamento rilevato. ● Cavo di rete Ethernet non inserito correttamente o difettoso. ● Dispositivo di rete (hub/switch) difettoso o non configurato correttamente.
	4 lampeggi, lungo spegnimento		Indirizzo IP doppio rilevato in rete. Per correggere questa situazione, provare ad assegnare un indirizzo IP diverso al controller Twido.
	6 lampeggi, lungo spegnimento		Uso di un indirizzo IP predefinito convertito valido; modalità FDR safe.
	9 lampeggi, lungo spegnimento		Guasto hardware Ethernet.

Messaggeria TCP Modbus

Panoramica

La messaggeria TCP Modbus consente al client Modbus TCP (controller master) di inviare e ricevere messaggi Ethernet verso e dal server Modbus TCP (controller slave). Visto che TCP Modbus è un protocollo di comunicazione peer-to-peer, un controller Twido che supporta Ethernet può essere sia client che server a seconda che stia rispettivamente interrogando o rispondendo a richieste.

Scambio di messaggi sulla rete Ethernet

La messaggeria Ethernet è gestita dall'istruzione EXCH3 e dal blocco funzione %MSG3: Anche l'instradamento verso un host Ethernet o tramite un gateway è supportato da EXCH3.

- **Istruzione EXCH3:** per trasmettere/ricevere messaggi.
- **Blocco funzione %MSG3:** per controllare gli scambi di messaggi.

Istruzione EXCH3

L'istruzione EXCH3 consente al controller Twido di inviare o ricevere informazioni da/a nodi di rete Ethernet. L'utente definisce una tabella di parole (%MWi:L) contenente le informazioni di controllo oltre ai dati da inviare e/o ricevere (fino a 128 byte in trasmissione e/o ricezione). Il formato delle tabelle di parole è descritto nella sezione seguente.
Uno scambio di messaggi viene eseguito utilizzando l'istruzione EXCH3.

Sintassi: [EXCH3 %MWi:L]

dove: L = numero di parole nelle tabelle di parole di comando, trasmissione e ricezione

Il controller Twido deve concludere lo scambio relativo alla prima istruzione EXCH3 prima di poter avviare una seconda istruzione di scambio. Nel caso di invio di più messaggi, utilizzare il blocco funzione %MSG3.
L'elaborazione dell'istruzione tramite lista EXCH3 avviene immediatamente, con qualsiasi trasmissione avviata per controllo dell'interrupt (anche la ricezione dei dati è controllata dall'interrupt), e viene considerata elaborazione in background.

Nota: l'uso dell'istruzione EXCH3 è identico all'uso di EXCHx (dove x = 1 or 2) con Modbus tradizionale. Anche la sintassi delle istruzioni è identica. Vi è tuttavia una differenza nelle informazioni veicolate dal byte 1 delle tabelle di trasmissione e ricezione. Mentre il byte 1 del Modbus tradizionale contiene l'indirizzo del collegamento seriale del controller slave, il byte 1 di TCP Modbus veicola il numero d'**indice** del controller client Modbus TCP. Il numero d'indice è specificato e memorizzato nella tabella Dispositivi remoti della finestra di dialogo Configurazione Ethernet di TwidoSoft (per ulteriori informazioni, vedere *Scheda Dispositivi remoti, p. 179*).

Tabella di parole EXCH3

Le dimensioni massime dei frame trasmessi e/o ricevuti è di 128 byte (si noti che questa limitazione si applica solo al client TCP Modbus, mentre il server TCP Modbus supporta la lunghezza standard Modbus PDU di 256 byte). Inoltre, la tabella delle parole associata all'istruzione EXCH3 è composta dalle tabelle di controllo, di trasmissione e ricezione, come descritto di seguito:

	Byte più significativo	**Byte meno significativo**
Tabella di controllo	Comando	Lunghezza (Trasmissione/Ricezione)
	Offset ricezione	Offset trasmissione
Tabella di trasmissione	Byte trasmesso 1. **Indice** come specificato nella tabella Dispositivi remoti della finestra di dialogo Configurazione Ethernet di TwidoSoft.	Byte trasmesso 2 come Modbus seriale
	...	Byte trasmesso n
	Byte trasmesso n+1	
Tabella di ricezione	Byte ricevuto 1. **Indice** come specificato nella tabella Dispositivi remoti della finestra di dialogo Configurazione Ethernet di TwidoSoft.	Byte ricevuto 2 come Modbus seriale
	...	Byte ricevuto p
	Byte ricevuto p+1	

Blocco funzione %MSG3

L'uso della funzione %MSG3 è identico a quello di %MSGx con Modbus tradizionale. %MSG3 consente di gestire gli scambi di dati fornendo:

- Controllo degli errori di comunicazione
- Coordinamento di più messaggi
- Trasmissione di messaggi di priorità

Il blocco funzione %MSGx dispone di un ingresso e due uscite associati:

I/O	**Definizione**	**Descrizione**
R	Azzera ingresso	Impostato a 1: reinizializza la comunicazione o azzera il blocco (%MSGx.E = 0 e %MSGx.D = 1).
%MSGx.D	Comunicazione completata	0: richiesta in corso. 1: comunicazione eseguita se viene ricevuto il segnale di fine trasmissione, carattere finale, errore o azzeramento del blocco.
%MSGx.E	errore	0: lunghezza messaggio corretta e collegamento corretto. 1: se il comando è sbagliato, se la tabella è configurata in modo non corretto, se viene ricevuto un carattere errato (velocità, parità, ecc.) o se la tabella di ricezione è piena.

Codice di errore EXCH3

Quando si verifica un errore con l'istruzione EXCH3:

- i bit **%MSG3.D** e **%MSG3.E** sono impostati a **1** e
- il **codice di errore** della comunicazione Ethernet è registrato nella parola di sistema **%SW65**.

La tabella seguente presenta il codice di errore EXCH3:

Codice di errore EXCH3 (registrato nella parola di sistema %SW65)
Codici di errore standard comuni a tutte le istruzioni EXCHx (x = 1, 2, 3): 0 - Operazione riuscita 1 – numero di bit da trasmettere troppo grande (> 128) 2 - Tabella di trasmissione troppo piccola 3 - Tabella di parole troppo piccola 4 - Overflow tabella di ricezione 5 - Timeout scaduto (Si noti che il codice di errore 5 è vuoto con l'istruzione EXCH3 e viene sostituito dai codici di errore 109 e 122 specifici di Ethernet descritti di seguito). 6 - Trasmissione 7 - Comando errato nella tabella 8 - Porta selezionata non configurata/disponibile 9 – Errore di ricezione 10 - Impossibile utilizzare %KW se in ricezione 11 - Offset di trasmissione più grande della tabella di trasmissione 12 - Offset di ricezione più grande della tabella di ricezione 13 - Interruzione dell'elaborazione EXCH da controller
Codici di errore specifici di Ethernet per EXCH3: 101 - Indirizzo IP non esistente 102 - Connessione TCP interrotta 103 - Nessun socket disponibile (tutti i canali di connessione sono occupati) 104 - Interruzione di rete 105 - Impossibile raggiungere la rete 106 - Interruzione di rete al reset 107 - Connessione terminata dal dispositivo peer 108 - Connessione ripristinata dal dispositivo peer 109 - Timeout di connessione scaduto 110 - Tentativo di connessione rifiutato 111 - Host spento 120 - Indice sconosciuto (il dispositivo remoto non è indicizzato nella tabella di configurazione) 121 - Irreversibile (MAC, Chip, IP doppio)122 - Timeout di ricezione scaduto dopo l'invio dei dati 123 - Inizializzazione Ethernet in corso

Incorporate, funzioni analogiche

7

In breve

Oggetto di questo capitolo

Questo capitolo descrive in che modo gestire il canale e i potenziometri analogici incorporati.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Potenziometro analogico	190
Canale analogico	192

Potenziometro analogico

Introduzione

I controller Twido dispongono di:
- un potenziometro analogico sui controller TWDLC•A10DRF, TWDLC•A16DRF e su tutti i controller modulari (TWDLMDA20DTK, TWDLMDA20DUK, TWDLMDA20DRT, TWDLMDA40DTK e TWDLMDA40DUK),
- Due potenziometri sui controller TWDLC•A24DRF e TWDLCA•40DRF.

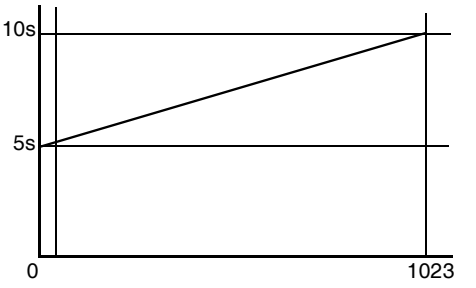
Programmazione

I valori numerici, compresi tra 0 e 1023 per il potenziometro analogico 1 e tra 0 e 511 per il potenziometro analogico 2, e corrispondenti ai valori analogici forniti da tali potenziometri, sono contenuti nelle due parole di ingresso seguenti:
- %IW0.0.0 per il potenziometro analogico 1 (posto a sinistra)
- %IW0.0.1 per il potenziometro analogico 2 (posto a destra)

È possibile utilizzare queste due parole nelle operazioni aritmetiche. Possono essere utilizzate per qualsiasi tipo di regolazione, ad esempio, preimpostare una temporizzazione o un contatore, regolare la frequenza del generatore di impulsi o il tempo di preriscaldamento della macchina.

Esempio

Regolazione della durata della temporizzazione da 5 a 10 con il potenziometro 1:

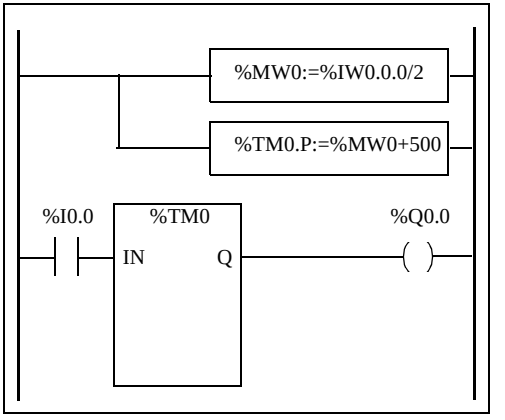
Per questa regolazione, viene utilizzato in pratica l'intero intervallo di regolazione del potenziometro 1, da 0 a 1023.	

I parametri che seguono vengono selezionati nella configurazione del blocco temporizzazione %TM0:

- Tipo TON
- Base tempo (TB): 10 ms

Il valore preimpostato della temporizzazione viene calcolato dal valore di regolazione del potenziometro tramite l'equazione seguente %TM0.P := (%IW0.0.0/2)+500..

Codice dell'esempio precedente:




```
LD      1
[%MW0:=%IW0.0.0/2]
[%TM0.P:=%MW0+500]
BLK     %TM0
LD      %I0.0
IN
OUT_BLK
LD      Q
ST      %Q0.0
END_BLK
.....................
```

Canale analogico

Introduzione

Tutti i controller modulari (TWDLMDA20DTK, TWDLMDA20DUK, TWDLMDA20DRT, TWDLMDA40DTK e TWDLMDA40DUK) presentano un canale analogico incorporato. La tensione in ingresso è compresa tra 0 e 10 V e tra 0 e 511 per il segnale digitalizzato. Il canale analogico sfrutta il semplice schema di calcolo della media su otto campioni..

Principio

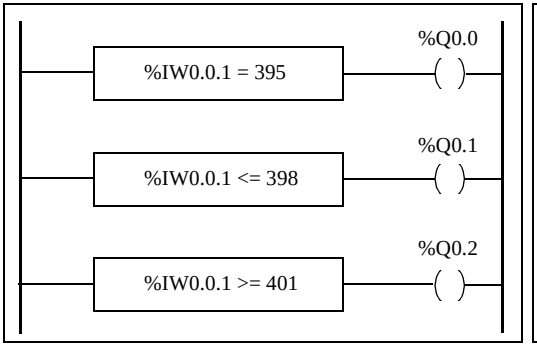
Un convertitore analogico-digitale campiona una tensione di ingresso tra 0 e 10 V con un valore digitale tra 0 e 511. Questo valore viene memorizzato nella parola di sistema %IW0.0.1. Il valore è lineare in tutto l'intervallo e ogni incremento è pari a 20 mV (10 V/512). Non appena il sistema rileva il valore 511, il canale è considerato saturato.

Esempio di programmazione

Controllo della temperatura di un forno: la temperatura di cottura è impostata a 350°C. Una variazione di +/- 2,5°C provoca l'apertura delle uscite %Q0.0 e %Q0.2, rispettivamente. In pratica, in questo esempio vengono utilizzati tutti i possibili intervalli di impostazione del canale analogico da 0 a 511. Le impostazioni analogiche dei Setpoint della temperatura sono:

Temperatura (°C)	Tensione	Parola di sistema %IW0.0.1
0	0	0
347,5	7,72	395
350	7,77	398
352,5	7,83	401
450	10	511

Codice dell'esempio precedente:




```
LD    [%IW0.0.1 = 395]
ST    %Q0.0

LD    [%IW0.0.1 <= 398]
ST    %Q0.1

LD    [%IW0.0.1 >= 401]
ST    %Q0.2
```

Gestione dei moduli analogici

8

In breve

Oggetto di questo capitolo

Questo capitolo descrive le procedure di gestione dei moduli analogici dei controller Twido.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Panoramica sui moduli analogici	194
Indirizzamento degli I/O analogici	195
Configurazione degli I/O analogici	196
Informazioni di stato del modulo analogico	202
Esempio di utilizzo dei moduli analogici	203

Panoramica sui moduli analogici

Introduzione

Oltre ai potenziometri a 10 bit e al canale analogico a 9 bit incorporati, tutti i controller Twido che supportano l'espansione degli I/O sono anche in grado di configurare e comunicare con i moduli I/O analogici.
I moduli analogici sono:

Nome	Canali	Gamma segnale	Codifica
TWDAMI2HT	2 In	0 - 10 Volt o 4 - 20 mA	12 Bit
TWDAMO1HT	1 Out	0 - 10 Volt o 4 - 20 mA	12 Bit
TWDAMM3HT	2 In, 1 Out	0 - 10 Volt o 4 - 20 mA	12 Bit
TWDALM3LT	2 In, 1 Out	0 - 10 V, ingressi Th o PT100, uscite da 4 a 20 mA	12 Bit
TWDAVO2HT	2 Out	+/- 10 Volt	11 Bit + segno
TWDAMI4LT	4 In	0 - 10 Volt, 0 - 20 mA, sensori a 3 fili NI o PT	12 Bit
TWDAMI8HT	8 In	0 - 10 Volt o 0 - 20 mA	10 bit
TWDARI8HT	8 In	Sensori NTC o PTC	10 bit

Funzionamento dei moduli analogici

Le parole in ingresso e in uscita (%IW e %QW) sono utilizzate per scambiare dati tra l'applicazione utente e i canali analogici. L'aggiornamento di queste parole avviene in modo sincronizzato con la scansione del controller in modalità RUN.

⚠ ATTENZIONE

AVVIO IMPREVISTO DEI DISPOSITIVI

Quando il controller è in STOP, l'uscita analogica si trova in posizione di sicurezza. Come con le uscite digitali, il setpoint predefinito è a zero.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

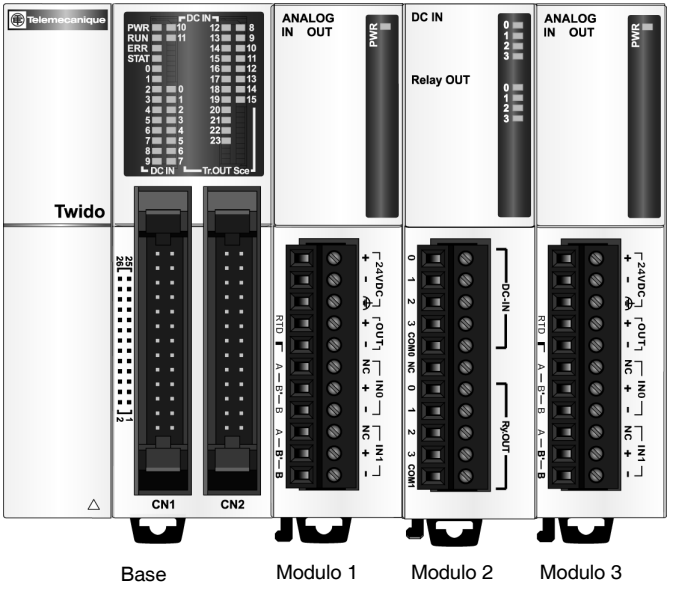
Indirizzamento degli I/O analogici

Introduzione

Gli indirizzi vengono assegnati ai canali analogici in base alla loro posizione nel bus di espansione.

Esempio di indirizzamento I/O analogico

In questo esempio, TWDLMDA40DUK ha un potenziometro incorporato a 10 bit regolato analogicamente e un canale analogico incorporato a 9 bit. Sul bus di espansione, sono configurati: un modulo analogico TWDAMM3HT, un modulo di relè digitale di I/O TWDDMM8DRT e un secondo modulo analogico TWDAMM3HT.



La tabella che segue mostra l'indirizzamento di ciascuna uscita.

Descrizione	Base	Modulo 1	Modulo 2	Modulo 3
Potenziometro 1	%IW0.0.0			
Canale analogico incorporato	%IW0.0.1			
Canale 1 d'ingresso analogico		%IW0.1.0		%IW0.3.0
Canale 2 d'ingresso analogico		%IW0.1.1		%IW0.3.1
Canale 1 di uscita analogico		%QW0.1.0		%QW0.3.0
Canali d'ingresso digitali			%I0.2.0 - %I0.2.3	
Canali d'uscita digitali			%Q0.2.0 -%Q0.2.3	

Configurazione degli I/O analogici

Introduzione

Questa sezione fornisce informazioni sulla configurazione degli ingressi e delle uscite del modulo analogico.

Configurazione degli I/O analogici

La finestra di dialogo **Configurazione del modulo** consente di gestire i parametri dei moduli analogici.
Per accedervi, utilizzare il browser dell'applicazione o il menu **Hardware**.

Nel browser dell'applicazione	Nel menu Hardware
1. Selezionare un modulo.	1. Selezionare **Configura modulo**.
2. Fare clic con il pulsante destro del mouse su **Configura** per aprire direttamente la finestra di dialogo **Configurazione del modulo - (Posizione e rif. modulo)**.	2. Selezionare un modulo dalla finestra di dialogo **Configurazione del modulo - Scelta del modulo**.
	3. Impostare i parametri nella finestra di dialogo **Configurazione del modulo - (Posizione e rif. modulo)** che viene visualizzata.

Nota: quando non si è collegati a un controller, è possibile modificare solo i parametri non in linea.

Barra del titolo e contenuti

La barra del titolo visualizza il riferimento del modulo e la sua posizione sul bus di espansione.
La parte superiore della finestra di dialogo mostra un'area di **Descrizione**.
La tabella mostra: **Indirizzo**, **Simbolo**, **Tipo**, **Intervallo**, **Minimo**, **Massimo** e **Unità**

- In TWDAMI4LT e TWIDAMI8HT, la tabella è preceduta dalla casella di riepilogo **Tipo di ingresso**.
- In TWDAVO2HT e TWDAMI8HT, la colonna **Tipo** è sostituita dalla colonna **Utilizzato** con caselle di controllo.
- In TWDARI8HT, ogni canale (0-7) è configurato singolarmente in una scheda in cui è possibile scegliere il metodo di configurazione **Grafico** o **Formula**. La tabella è visualizzabile nella scheda **Ricap**.

Descrizione

Il campo **Descrizione** descrive brevemente questo modulo.

Indirizzo

Ogni riga del foglio elettronico rappresenta un canale di ingresso o di uscita del modulo.
Gli indirizzi di ognuno di essi sono identificati nella tabella seguente, in cui "i" indica la posizione del modulo sul bus di espansione.

Nome del modulo	Indirizzo
TWDALM3LT	2 ingressi (%IWi.0, %IWi.1), 1 uscita (%QWi.0)
TWDAMM3HT	2 ingressi (%IWi.0, %IWi.1), 1 uscita (%QWi.0)
TWDAMI2HT	2 ingressi (%IWi.0, %IWi.1)
TWDAMO1HT	1 uscita (%QWi.0)
TWDAVO2HT	2 uscite (%QWi.0, %QWi.1)
TWDAMI4LT	4 ingressi (da %IWi.0 a %IWi.3)
TWDAMI8HT	8 ingressi (da %IWi.0 a %IWi.7)
TWDARI8HT	8 ingressi (da %IWi.0 a %IWi.7)

Simbolo

Visualizzazione in sola lettura dell'eventuale simbolo assegnato all'indirizzo.

Tipo ingresso e/ o tipo

Identifica la modalità di un canale. Le scelte dipendono dal canale e dal tipo di modulo.
Per TWDAMO1HT, TWDAMM3HT e TWDALM3LT, è possibile configurare il tipo di canale di uscita singolo in modo che risulti:

Tipo
Non utilizzato
0 - 10 V
4 – 20 mA

Per TWDAMI2HT e TWDAMM3HT, è possibile configurare i due tipi di canale di ingresso in modo che risultino:

Tipo
Non utilizzato
0 - 10 V
4 – 20 mA

Per TWDALM3LT è possibile configurare i due tipi di canale di ingresso in modo che risultino:

Tipo
Non utilizzato
Termocoppia K
Termocoppia J
Termocoppia T
PT 100

Per TWDAVO2HT, non è disponibile alcun tipo da regolare.
Per TWDAMI4LT è possibile configurare i quattro tipi di ingresso in modo che risultino:

Tipo d'ingresso	**Tipo**
Tensione	Non utilizzato 0-10 V
Corrente	Non utilizzato 0-20 mA
Temperatura	Non utilizzato PT 100 PT 1000 NI 100 NI 1000

Per TWDAMI8HT è possibile configurare gli otto tipi di ingresso in modo che risultino:

Tipo d'ingresso
0 - 10 V
0 - 20 mA

Per TWDARI8HT è possibile configurare singolarmente ciascun canale di ingresso (0-7) dal campo **Funzionamento** nella parte inferiore della finestra. Se necessario, selezionare direttamente una **Modalità** e un **Intervallo**. È quindi possibile visualizzare un riepilogo di tutte le informazioni nella scheda Ricap, dove la colonna **Tipo** mostra:

Tipo
Non utilizzato
NTC / CTN
PTC / CTP

⚠ ATTENZIONE

DANNI ALLE APPARECCHIATURE

Se l'ingresso è stato collegato in base a una misura di tensione e si configura TwidoSoft per il tipo corrente di configurazione, il modulo analogico potrebbe subire danni permanenti. Verificare che il cablaggio sia conforme alla configurazione di TwidoSoft.

La mancata osservanza di questa precauzione può causare ferite o danni alle apparecchiature.

Intervallo Identifica l'intervallo di valori di un canale. Le scelte dipendono dal tipo specifico di canale e di modulo.
Dopo aver configurato il **Tipo**, è possibile impostare l'**Intervallo** corrispondente. Una tabella mostra i valori **Minimo** e **Massimo** accettati (fissi o definiti dall'utente) oltre alle eventuali **Unità**.

Intervallo (Sensori NTC)	Minimo	Massimo	Unità	Moduli analogici I/O
Normale	0	4095	Nessuna	TWDALM3LT TWDAMO1HT TWDAMM3HT TWDAMI2HT TWDAMI4LT
	-2048	2047		TWDAVO2HT
	0	1023		TWDAMI8HT TWDARI8HT
Personalizzato	Definito dall'utente con min. di -32768	Definito dall'utente con max di 32767	Nessuna	Tutti i moduli analogici I/O
Celsius	-1000	5000	0,1°C	TWDALM3LT
	Dinamicamente aggiornato da TwidoSoft secondo i parametri definiti dall'utente.			TWDARI8HT
	-2000	6000		TWDAMI4LT (sensori Pt)
	-500	1500		TWDAMI4LT (sensori Ni)
Fahrenheit	-1480	9320	0,1°F	TWDALM3LT
	Dinamicamente aggiornato da TwidoSoft secondo i parametri definiti dall'utente.			TWDARI8HT
	-3280	11120		TWDAMI4LT (sensori Pt)
	-580	3020		TWDAMI4LT (sensori Ni)
Resistenza	100	10000	Ohm	TWDARI8HT
	74	199		TWDAMI4LT (Ni100)
	742	1987		TWDAMI4LT (Ni1000)
	18	314		TWDAMI4LT (Pt100)
	184	3138		TWDAMI4LT (Pt1000)

Metodo grafico o formula

In TWDARI8HT, ciascun canale (0-7) viene configurato singolarmente in una scheda. Selezionare la casella **Utilizzato** quindi scegliere tra i metodi di configurazione **Grafico** e **Formula**.

- **Metodo grafico**
  (**R1**, **T1**) e (**R2**, **T2**) corrispondono alle coordinate in formato a virgola mobile di due punti appartenenti alla curva.
  R1(predefinito 8700) e **R2** (predefinito 200) sono valori espressi in Ohm.
  Per i valori **T1** (predefinito 233,15) e **T2** (predefinito 398,15) le unità possono essere impostate nella casella di riepilogo **Unità**: **Kelvin** (predefinito), **Celsius** o **Fahrenheit**.
  Nota: cambiando l'unità della temperatura dopo aver impostato i valori T1 e T2 non provoca il ricalcolo automatico dei valori T1 e T2 nella nuova unità.
- **Metodo formula**
  Se si conoscono i parametri **Rref**, **Tref** e **B**, è possibile utilizzare questo metodo per definire le caratteristiche del sensore.
  Rref (predefinito 330) è espresso in Ohm.
  B è predefinito a 3569 (min. 1, max. 32767).
  Per **Tref** (val. predefinito 298,15) è possibile impostare l'unità nella casella di riepilogo **Unità**: **Kelvin** (predefinito), **Celsius** o **Fahrenheit**.

Di seguito viene fornita una tabella di valori di **Tref** min/max corrispondenti tra le unità:

Unità	Valore min.	Valore max.
Kelvin	1	650
Celsius	-272	376
Fahrenheit	-457	710

Nelle finestre Grafico e Formula, è possibile importare i valori da un altro canale nel canale correntemente configurato:

1. Selezionare un numero di canale nella casella **N. canale**.
2. Premere il pulsante **Importa valori**.

A queste finestre possono essere associati messaggi di errore o avviso.

Nota: se si inizia a impostare i valori e si decide successivamente di passare da Grafico a Formula o viceversa, il sistema visualizza un messaggio di avviso che indica che verranno ripristinati i valori predefiniti e che i valori modificati verranno persi.

Informazioni di stato del modulo analogico

Tabella di stato La tabella che segue include le informazioni necessarie per controllare lo stato dei moduli I/O analogici.

Parola di sistema	Funzione	Descrizione
%SW80	Stato I/O base	Bit [0] Canali in funzionamento normale (per tutti i canali). Bit [1] Modulo in inizializzazione (oppure informazioni di inizializzazione per tutti i canali). Bit [2] Guasto hardware (guasto alimentazione esterna, comune a tutti i canali) Bit [3] Errore di configurazione del modulo Bit [4] Conversione della via d'ingresso dei dati 0 in corso Bit [5] Conversione della via d'ingresso dei dati 1 in corso Bit [6] Ingresso termocoppia canale 0 non configurato Bit [7] Ingresso termocoppia canale 1 non configurato Bit [8] Non utilizzato Bit [9] Non utilizzato Bit [10] Ingresso analogico dati canale 0 al di sopra dell'intervallo Bit [11] Ingresso analogico dati canale 1 al di sopra dell'intervallo Bit [12] Collegamento errato (canale dei dati d'ingresso analogico 0 inferiore all'intervallo corrente, loop di corrente aperto) Bit [13] Collegamento errato (canale dei dati d'ingresso analogico 1 inferiore all'intervallo corrente, loop di corrente aperto) Bit [14] Non utilizzato Bit [15] Canale di uscita non disponibile
%SW81	Stato modulo di espansione I/O 1: Definizioni come %SW80	
%SW82	Stato modulo di espansione I/O 2: Definizioni come %SW80	
%SW83	Stato modulo di espansione I/O 3: Definizioni come %SW80	
%SW84	Stato modulo di espansione I/O 4: Definizioni come %SW80	
%SW85	Stato modulo di espansione I/O 5: Definizioni come %SW80	
%SW86	Stato modulo di espansione I/O 6: Definizioni come %SW80	
%SW87	Stato modulo di espansione I/O 7: Definizioni come %SW80	

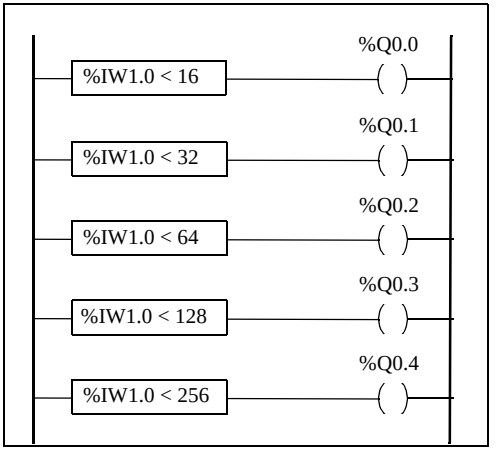
Esempio di utilizzo dei moduli analogici

Introduzione

Questa sezione fornisce un esempio di utilizzo dei moduli analogici disponibili con Twido.

Esempio: ingresso analogico

Questo esempio confronta il segnale di ingresso analogico con cinque valori soglia separati. Viene eseguito un confronto dell'ingresso analogico e viene impostato un bit sul controller della base, se inferiore o uguale al valore soglia.




```
LD [%IW1.0 < 16]
ST %Q0.0

LD [%IW1.0 < 32]
ST %Q0.1

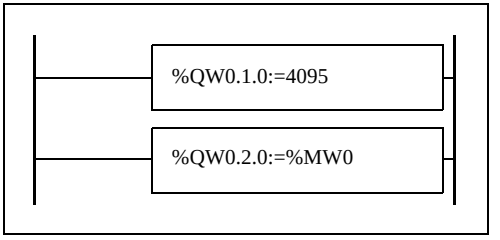
LD [%IW1.0 < 64]
ST %Q0.2

LD [%IW1.0 < 128]
ST %Q0.3

LD [%IW1.0 < 256]
ST %Q0.4
```

Esempio: uscita analogica

In questo programma si utilizza una scheda analogica negli alloggiamenti 1 e 2. La scheda utilizzata nell'alloggiamento 1 ha un'uscita a 10 Volt con una serie "normale":




```
LD   1
[%QW0.1.0:=4095
LD   1
[%QW0.2.0:=%MW0
```

- Esempio di valori di uscita per %QW1.0=4095 (caso normale):

La tabella che segue fornisce il valore della tensione di uscita in base al valore massimo attribuito a %QW1.0:

	valore numerico	**valore analogico (volt)**
Minimo	0	0
Massimo	4095	10
Valore 1	100	0,244
Valore 2	2460	6

- Esempio dei valori delle uscite per una serie personalizzata (minimo =0, massimo =1000):

La tabella che segue fornisce il valore della tensione di uscita in base al valore massimo attribuito a %QW1.0:

	valore numerico	**valore analogico (volt)**
Minimo	0	0
Massimo	1000	10
Valore 1	100	1
Valore 2	600	6

Implementazione del bus AS-Interface V2

9

In breve

Oggetto di questo capitolo

Questo capitolo fornisce informazioni sull'implementazione software del modulo master AS-Interface TWDNOI10M3 e dei relativi slave.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Presentazione del bus AS-Interface V2	206
Descrizione funzionale generale	207
Principi di implementazione software	210
Descrizione della schermata di configurazione del bus AS-Interface	211
Configurazione del bus AS-Interface	213
Descrizione della schermata di debug	219
Modifica dell'indirizzo di uno slave	222
Aggiornamento della configurazione del bus AS-Interface in modalità collegato	224
Indirizzamento automatico di uno slave AS-Interface V2	229
Inserimento di un'apparecchiatura slave in una configurazione AS-Interface V2 esistente	230
Sostituzione automatica di uno slave AS-Interface V2 in errore	231
Indirizzamento degli ingressi/uscite associati alle apparecchiature slave collegate sul bus AS-Interface V2	232
Programmazione e diagnostica del bus AS-Interface V2	234
Modalità di funzionamento del modulo interfaccia bus AS-Interface V2	238

Presentazione del bus AS-Interface V2

Introduzione

Il bus AS-Interface (Actuator Sensor-Interface) permette l'interconnessione, su un unico cavo, di sensori e attuatori al livello più basso di automazione.
I sensori/attuatori sono definiti nella documentazione come **apparecchiature slave**.

Per implementare AS-Interface, è necessario definire il contesto hardware dell'applicazione in cui sarà integrato (bus di espansione, alimentazione, processore, moduli, apparecchiature slave AS-Interface collegate sul bus) e assicurare l'implementazione software.

Questo secondo aspetto sarà realizzato con i vari editor di TwidoSoft:
- in modalità locale
- o in modalità online.

Bus AS-Interface V2

Il modulo master AS-interface **TWDNOI10M3** integra le seguenti funzionalità:
- Profilo M3: questo profilo copre tutte le funzionalità definite dallo standard AS-Interface V2, ma non supporta i profili analogici S7-4.
- Un canale AS-Interface per modulo.
- Autoindirizzamento dello slave all'indirizzo 0.
- Gestione dei profili e dei parametri.
- Protezione contro l'inversione di polarità sugli ingressi del bus

Il bus AS-Interface prevede quindi:
- fino a 31 slave di tipo indirizzamento standard e 62 di tipo indirizzamento esteso,
- fino a 248 ingressi e 186 uscite,
- fino a 7 slave analogici (4 I/O max per slave),
- un tempo di ciclo di 10 ms massimo.

Su un controller modulare Twido o un controller compatto TWDLC•A24DRF o TWDLCA•40DRF è possibile collegare al massimo 2 moduli master AS-Interface.

Descrizione funzionale generale

Presentazione generale

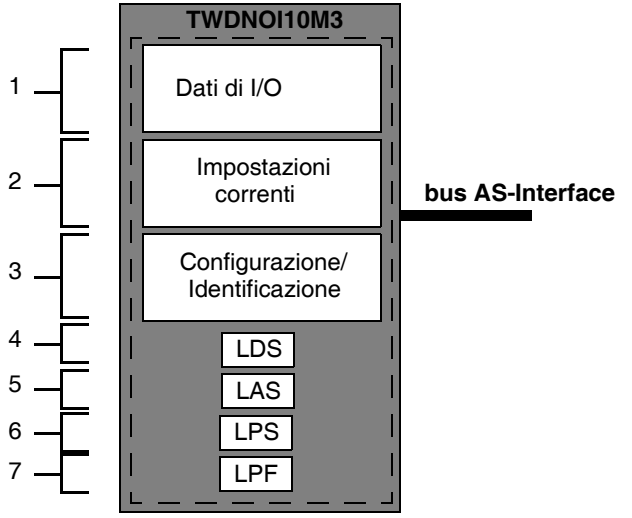
Per la configurazione di AS-Interface, il software TwidoSoft permette all'utente le seguente operazioni:

- configurazione del bus (dichiarazione degli slave e attribuzione degli indirizzi sul bus) in modo manuale,
- adattamento della configurazione rispetto alla situazione presente sul bus,
- acquisizione dei parametri degli slave,
- controllo dello stato del bus.

A questo scopo, tutte le informazioni provenienti dal o destinate al master AS-Interface vengono memorizzate in oggetti (parole e bit) specifici.

Struttura del master AS-Interface

Il modulo AS-Interface integra i campi di dati che permettono di gestire gli elenchi di slave e le immagini dei dati degli ingressi/uscite. Queste informazioni sono memorizzate in una memoria volatile.
La figura seguente presenta l'architettura del modulo **TWDNOI10M3**.



Legenda:

Indirizzo	Elemento	Descrizione
1	Dati di I/O (IDI, ODI)	Immagini dei 248 ingressi e delle 186 uscite del bus AS-Interface V2.
2	Parametri attuali (PI, PP)	Immagine dei parametri di tutti gli slave.

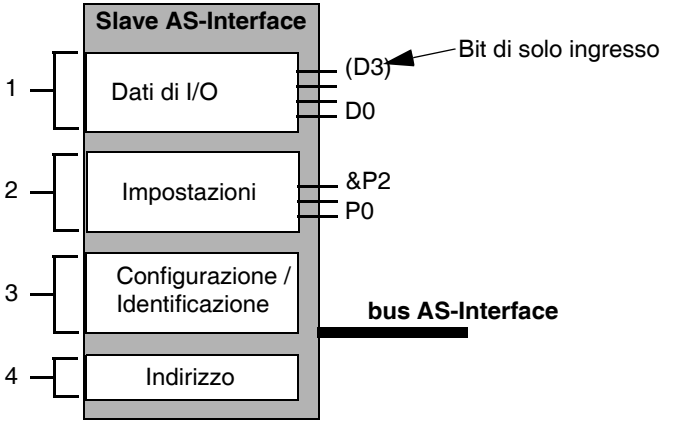
Indirizzo	Elemento	Descrizione
3	Configurazione/Identificazione (CDI, PCD)	Questo campo contiene tutti i codici di I/O e i codici di identificazione di tutti gli slave rilevati.
4	LDS	Lista di tutti gli slave rilevati sul bus.
5	LAS	Lista degli slave attivati sul bus.
6	LPS	Lista degli slave previsti sul bus e configurati tramite TwidoSoft.
7	LPF	Elenco degli slave con un errore periferica.

Struttura delle apparecchiature slave

Ognuno degli slave a indirizzamento standard dispone di:
- 4 bit di ingresso/uscita,
- 4 bit di parametrizzazione.

Ognuno degli slave a indirizzamento esteso dispone di:
- 4 bit di ingresso/uscita (ultimo bit riservato esclusivamente all'ingresso),
- 3 bit di parametrizzazione.

Ogni slave dispone di un proprio indirizzo, un profilo e un sottoprofilo (definizione dello scambio delle variabili).
La figura seguente presenta la struttura di uno slave a indirizzamento esteso.



Legenda:

Indirizzo	Elemento	Descrizione
1	Dati degli ingressi/uscite	I dati degli ingressi sono memorizzati dallo slave e messi a disposizione del master AS-Interface. I dati delle uscite sono aggiornati dal modulo master.
2	Impostazioni	I parametri permettono il controllo e la commutazione delle modalità di funzionamento interno del sensore o dell'attuatore.
3	Configurazione/ Identificazione	Il campo include: ● il codice corrispondente alla configurazione degli ingressi/uscite (I/O), ● il codice di identificazione dello slave (ID). ● i sottocodici di identificazione dello slave (ID1 e ID2).
4	Indirizzo	Indirizzo fisico dello slave.
Osservazione: i parametri di funzionamento, indirizzo, dati di configurazione e di identificazione sono salvati in una memoria non volatile.		

Principi di implementazione software

In breve

Per rispettare l'approccio utilizzato in TwidoSoft, nella creazione di un'applicazione AS-Interface l'utente deve procedere per passi successivi.

Principio di implementazione

L'utente deve sapere come configurare in modo funzionale il bus AS-Interface (Vedi *Inserimento di un'apparecchiatura slave in una configurazione AS-Interface V2 esistente, p. 230*).
La seguente tabella descrive le diverse fasi dell'implementazione software del bus AS-Interface V2.

Modalità	Fase	Descrizione
Non in linea	Dichiarazione del modulo	Scelta dell'alloggiamento del modulo master AS-Interface TWDNOI10M3 sul bus d'espansione.
	Configurazione del canale del modulo	Scelta delle modalità "master"
	Dichiarazione delle apparecchiature slave	Scelta per ogni apparecchiatura: ● del numero di alloggiamento sul bus, ● del tipo di slave di indirizzamento standard od esteso.
	Convalida dei parametri di configurazione	Convalida a livello slave.
	Convalida globale dell'applicazione	Convalida del livello applicazione.
Locale o collegato	Simbolizzazione (opzionale)	Rappresentazione simbolica delle variabili associate alle apparecchiature slave.
	Programmazione	Programmazione della funzione AS-Interface V2.
In linea	Trasferimento	Trasferimento dell'applicazione nel controller.
	Debug	Debug dell'applicazione tramite: ● la schermata di debug che permette da una parte la visualizzazione degli slave (indirizzo, parametri) e dall'altra l'indirizzamento degli slave agli indirizzi desiderati. ● schermate di diagnostica che consentono di identificare gli errori.

Nota: la dichiarazione e l'eliminazione del modulo master AS-Interface sul bus d'espansione si svolge come per gli altri moduli di espansione. Se tuttavia vi sono due moduli master AS-Interface dichiarati sul bus di espansione, TwidoSoft non permette più di dichiararne altri.

Precauzioni prima della connessione

Prima di collegare in modo software il PC al controller e per evitare eventuali problemi di rilevamento:
- verificare che non vi siano slave presenti fisicamente sul bus all'indirizzo 0,
- verificare che non vi siano 2 slave presenti fisicamente allo stesso indirizzo.

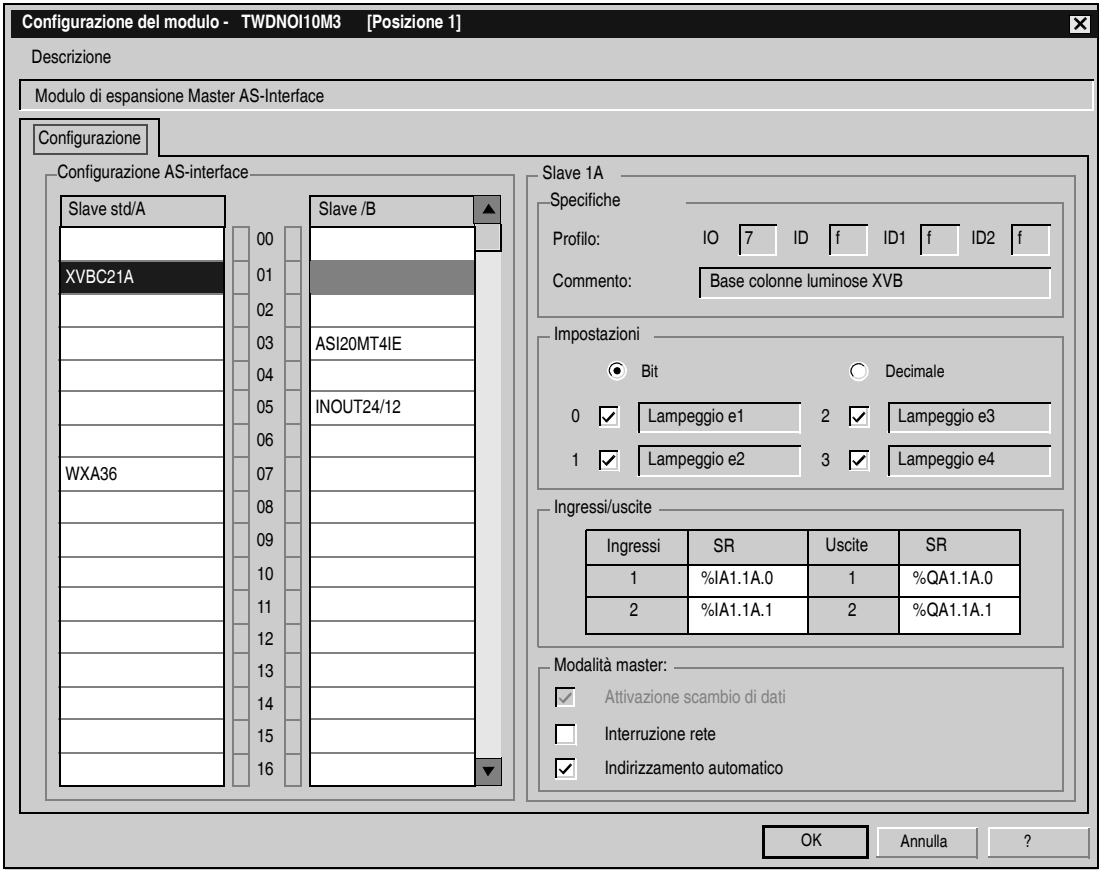
Descrizione della schermata di configurazione del bus AS-Interface

In breve

La schermata di configurazione del modulo master AS-Interface consente di accedere ai parametri associati al modulo e agli slave.
Permette la visualizzazione e la modifica dei parametri in modalità locale.

Illustrazione in modalità locale

Illustrazione della schermata di configurazione in modalità locale:



Descrizione della schermata di configurazione in modalità locale:

La schermata contiene tutte le informazioni relative al bus in tre gruppi distinti:

Blocchi	Descrizione
Configurazione AS-interface	Immagine del bus desiderato dall'utente: visualizzazione degli slave a indirizzamento standard e a indirizzamento esteso (previsti) sul bus. Scorrere verso il basso il cursore sulla barra verticale per accedere agli indirizzi successivi. Gli indirizzi in grigio corrispondono a indirizzi disponibili per la configurazione di uno slave. Se, ad esempio, viene dichiarato un nuovo slave standard all'indirizzo 1A, l'indirizzo 1B appare automaticamente in grigio.
Slave xxA/B	Configurazione dello slave selezionato: ● Caratteristiche: codice IO, codice ID, codici ID1 e ID2 (profili), e commento relativo alla slave, ● Parametri: elenco dei parametri (modificabili), in formato binario (4 caselle di selezione) o decimale (1 casella) in base alla scelta dell'utente, ● Ingressi/uscite: elenco di I/O disponibili e relativi indirizzi.
Modalità master	Possibilità di attivazione e disattivazione delle due funzionalità disponibili per questo modulo AS-Interface (ad esempio, dell'indirizzamento automatico). "Interruzione rete" permette di forzare il bus AS-Interface a entrare in modalità offline. Per impostazione predefinita, è selezionata la modalità "Indirizzamento automatico". Nota: la funzione "Attivazione scambio di dati" non è ancora disponibile.

La schermata propone inoltre 3 pulsanti:

Pulsanti	Descrizione
OK	consente di salvare la configurazione del bus AS-Interface visibile nella schermata di configurazione. Ritorna quindi alla schermata principale. La configurazione può essere trasferita al controller Twido.
Annulla	ritorna alla schermata principale senza salvare le modifiche effettuate.
?	apre la finestra della guida.

Nota: è possibile apportare modifiche nella schermata di configurazione solo in modalità locale.

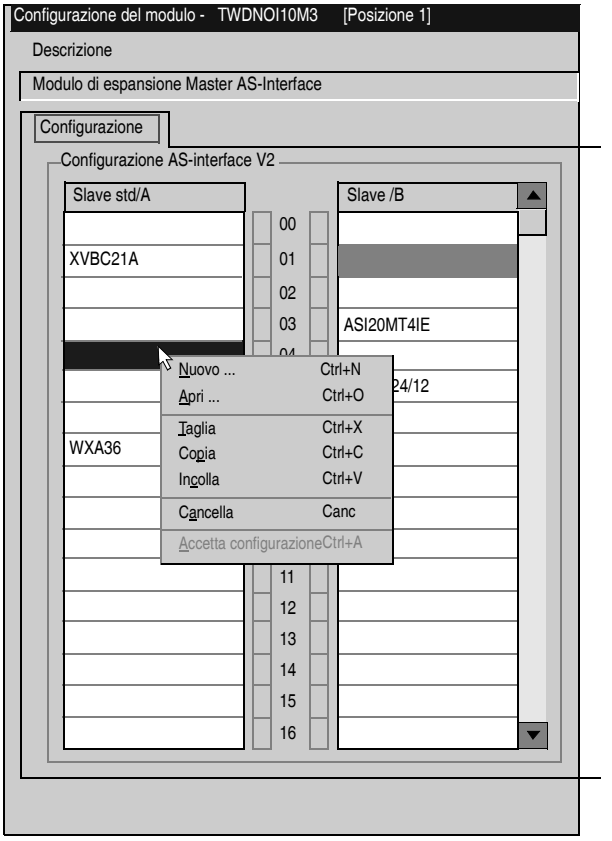
Configurazione del bus AS-Interface

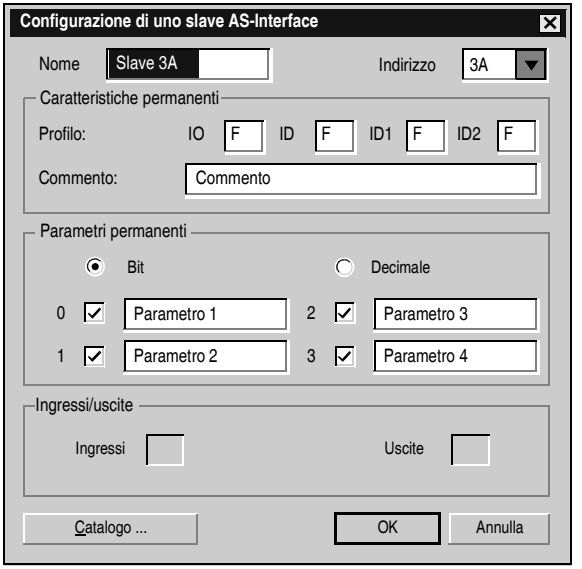
Introduzione

La configurazione del bus AS-Interface si esegue nella schermata di configurazione in modalità locale.
Dopo aver selezionato il master AS-Interface e le modalità master, la configurazione del bus AS-Interface consiste nella configurazione delle apparecchiature slave.

Procedura di dichiarazione e di configurazione di uno slave

Procedura per la creazione o la modifica di uno slave sul bus AS-Interface V2:

Passo	Azione
1	Sulla cella corrispondente all'indirizzo desiderato (non disattivata), nell'immagine del bus: ● Fare doppio clic: accesso alla fase 3 OPPURE ● Fare clic con il pulsante destro del mouse: Risultato:  Nota: viene visualizzato un menu contestuale. È possibile: ● configurare un nuovo slave sul bus; ● modificare la configurazione dello slave desiderato; ● copiare (in alternativa premere Ctrl+C), tagliare (in alternativa premere Ctrl+X) o incollare (in alternativa premere Ctrl+V) uno slave; ● eliminare (in alternativa premere Canc) uno slave.

Passo	Azione
2	Nel menu contestuale, selezionare: ● "Nuovo" per creare un nuovo slave: viene visualizzata una schermata di configurazione dello slave; il campo "Indirizzo" indica l'indirizzo selezionato, i campi di "Profilo" sono impostati in modo predefinito su F e tutti gli altri campi della schermata sono vuoti. ● "Apri" per creare un nuovo slave o per modificare la configurazione dello slave selezionato. In caso di un nuovo slave, viene visualizzata una nuova schermata di configurazione, il campo "Indirizzo" indica l'indirizzo selezionato, i campi di "Profilo" sono impostati in modo predefinito su F e tutti gli altri campi della schermata sono vuoti. In caso di modifica, la schermata di configurazione dello slave viene visualizzata con i campi contenenti i valori definiti in precedenza per lo slave selezionato. Illustrazione di una schermata di configurazione per un nuovo slave: 
3	Nella schermata di configurazione dello slave che viene visualizzata, è possibile immettere o modificare: ● il nome del nuovo profilo (max. 13 caratteri); ● un commento (opzionale); o fare clic sul pulsante "Catalogo..." e scegliere uno slave della famiglia di profili AS-Interface preconfigurati.

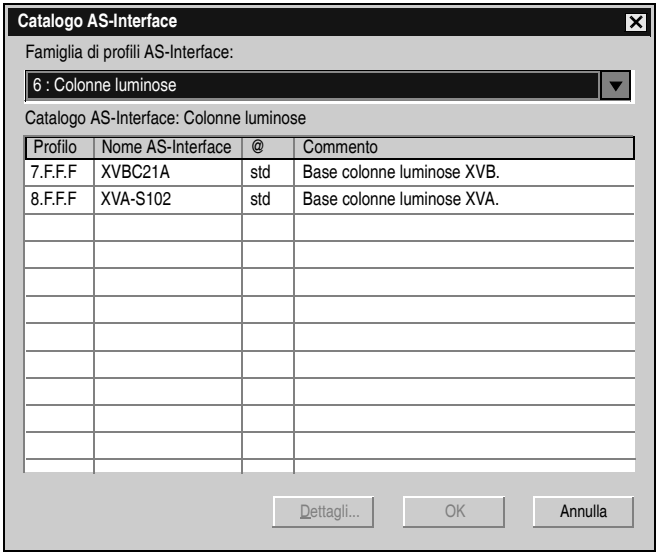
Passo	Azione
4	Immettere: ● il codice IO (corrispondente alla configurazione degli ingressi/uscite); ● il codice ID (identificatore) oltre a ID1 e ID2 per un tipo esteso. Nota: i campi "Ingressi" e "Uscite" indicano il numero di canale di ingresso e di uscita. Questi campi vengono implementati in modo automatico all'immissione del codice IO.
5	Per ogni parametro, definire: ● l'acquisizione da parte del sistema (casella selezionata per "Bit" o valore decimale compreso tra 0 e 15 per "Decimale"); ● una descrizione dettagliata del "Parametro X" (opzionale). Nota: i parametri selezionati sono l'immagine dei parametri permanenti da fornire al master AS-Interface.
6	Se necessario, modificare l'"Indirizzo" (nel limite degli indirizzi disponibili sul bus) facendo clic sulle frecce su e giù che si trovano a sinistra dell'indirizzo stesso (in questo caso si ha accesso agli indirizzi autorizzati) o immettendo direttamente l'indirizzo dalla tastiera.
7	Confermare la configurazione dello slave facendo clic su "OK". Il risultato è la verifica che: ● i codici IO e ID sono autorizzati; ● l'indirizzo dello slave è autorizzato (in caso di immissione dalla tastiera) secondo il codice ID (gli slave "banco"/B sono disponibili solo se il codice ID è uguale a A). In caso di errore, un messaggio avverte l'utente del tipo di errore (ad esempio, il messaggio informa che il tipo di indirizzo non è consentito per lo slave) e la schermata viene visualizzata nuovamente con i valori iniziali (nel profilo o nell'indirizzo, a seconda dell'errore).

Nota: il software limita il numero di dichiarazione dello slave analogico a 7.

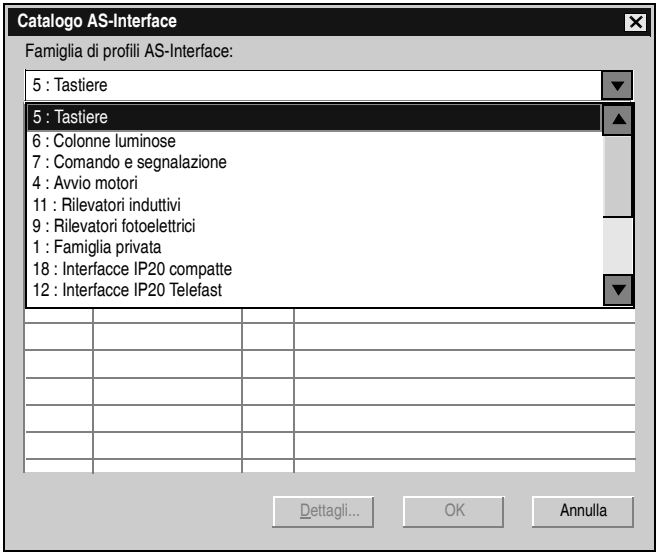
Nota: per quanto riguarda il catalogo Schneider AS-Interface, facendo clic sul pulsante Catalogo è possibile creare e configurare degli slave in "Famiglia privata" (diversi da quelli del catalogo Schneider AS-Interface).

Catalogo AS-Interface

Il pulsante catalogo serve a facilitare la configurazione degli slave sul bus. Utilizzare questo pulsante quando si utilizza uno slave della famiglia Schneider per eseguire la configurazione in modo semplice e rapido.
Facendo clic sul pulsante "Catalogo" della finestra "Configurazione di uno slave AS-Interface", viene visualizzata la finestra seguente:



Nel menu a discesa si può accedere a tutte le famiglie del catalogo AS-Interface Schneider:



Una volta scelta la famiglia, viene visualizzata la lista degli slave corrispondenti. Fare clic sullo slave desiderato e confermare facendo clic su "Ok".

Nota: per visualizzare le caratteristiche di uno slave, fare clic sul pulsante "Dettagli".

Nota: è possibile aggiungere e configurare degli slave che non fanno parte del catalogo Schneider. A questo scopo, è sufficiente scegliere la famiglia privata e configurare il nuovo slave.

Descrizione della schermata di debug

In breve

Quando il PC è **collegato** al controller (dopo aver caricato l'applicazione nel controller), a destra della scheda "Configurazione" viene visualizzata la scheda "Debug", che consente l'accesso alla schermata di debug.
La schermata di debug fornisce un'immagine dinamica del bus fisico che comprende:

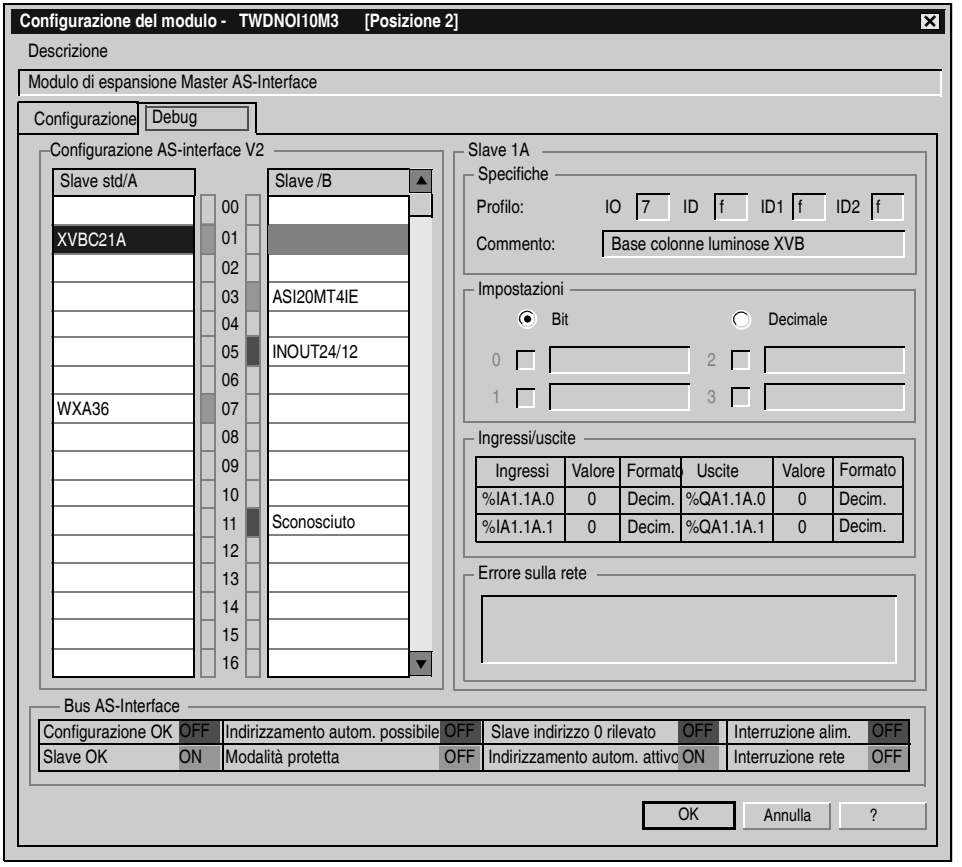
- l'elenco degli slave previsti (specificati) durante la configurazione, con il relativo nome e l'elenco degli slave rilevati (sconosciuti se non previsti);
- lo stato del modulo AS-Interface e delle apparecchiature slave;
- l'immagine del profilo, dei parametri e dei valori di ingressi/uscite degli slave selezionati.

Consente inoltre all'utente di:

- ottenere una diagnostica degli slave in errore (Vedi *Visualizzazione degli stati degli slave, p. 221*);
- modificare l'indirizzo di uno slave in modalità Collegato (Vedi *Modifica dell'indirizzo di uno slave, p. 222*);
- trasmettere l'immagine degli slave alla schermata di configurazione (Vedi *Aggiornamento della configurazione del bus AS-Interface in modalità collegato, p. 224*);
- assegnare a tutti gli slave gli indirizzi desiderati (durante il primo debug).

Illustrazione della schermata "Debug"

L'illustrazione della schermata di debug (solo in modalità Collegato) è visualizzata di seguito:



Descrizione della schermata di debug

La schermata di "Debug" fornisce le stesse indicazioni della schermata di configurazione (Vedi *Descrizione della schermata di configurazione in modalità locale:, p. 212*).
Le differenze sono elencate nella tabella seguente.

Blocco	Descrizione
Configurazione AS-interface V2	Immagine del bus fisico Comprende la visualizzazione dello stato degli slave: ● indicatore verde: lo slave a questo indirizzo è attivo. ● indicatore rosso: lo slave a questo indirizzo è in errore e un messaggio informa del tipo di errore nella finestra "Errore sulla rete".
Slave xxA/B	Immagine della configurazione dello slave selezionato: ● Caratteristiche: immagine del profilo rilevato (in grigio, non modificabile). ● Parametri: immagine dei parametri rilevati. L'utente può solo selezionare il formato di visualizzazione dei parametri. ● Ingressi/uscite: i valori degli ingressi/uscite rilevati vengono visualizzati ma non possono essere modificati.
Errore sulla rete	Se lo slave selezionato è in errore, informa del tipo di errore.
Bus AS-Interface	Informazioni derivanti da un comando implicito "Read Status". ● indica lo stato del bus: ad esempio, "Configurazione OK = OFF" indica che la configurazione prevista dall'utente non corrisponde a quella fisica del bus; ● indica le funzionalità consentite per il modulo master AS-Interface: ad esempio "Indirizzamento automatico attivo = ON" indica che la modalità master di indirizzamento automatico è autorizzata.

Visualizzazione degli stati degli slave

Se l'indicatore associato a un indirizzo è rosso, lo slave correlato a tale indirizzo è in errore. La finestra "Errore sulla rete" fornisce quindi la diagnostica dello slave selezionato.
Descrizione degli errori:

- il profilo previsto dall'utente in configurazione a un dato indirizzo non corrisponde al profilo reale individuato a questo indirizzo sul bus (diagnostica: "Errore di profilo");
- un nuovo slave non previsto in configurazione viene rilevato sul bus: viene visualizzato un indicatore rosso per questo indirizzo e il nome dello slave "Sconosciuto" (diagnostica: "Slave non progettato");
- errore di periferica, se lo slave rilevato lo supporta (diagnostica: "Errore periferica");
- in configurazione è previsto un profilo, ma non viene rilevato alcuno slave a questo indirizzo sul bus (diagnostica: "Slave non rilevato").

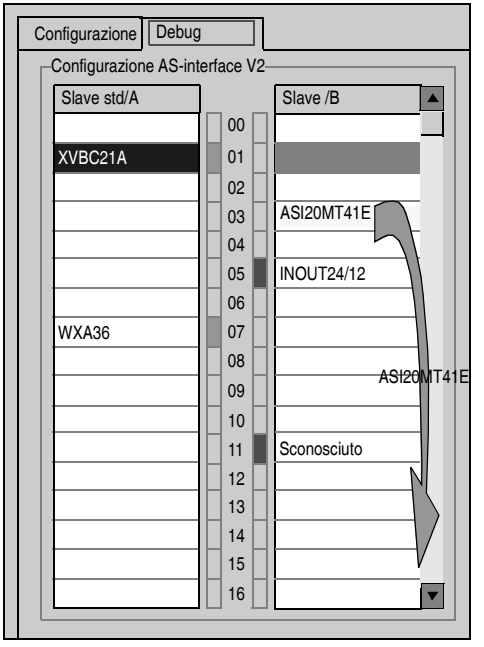
Modifica dell'indirizzo di uno slave

In breve

Mediante la schermata di debug, l'utente può modificare l'indirizzo di uno slave in modalità collegato.

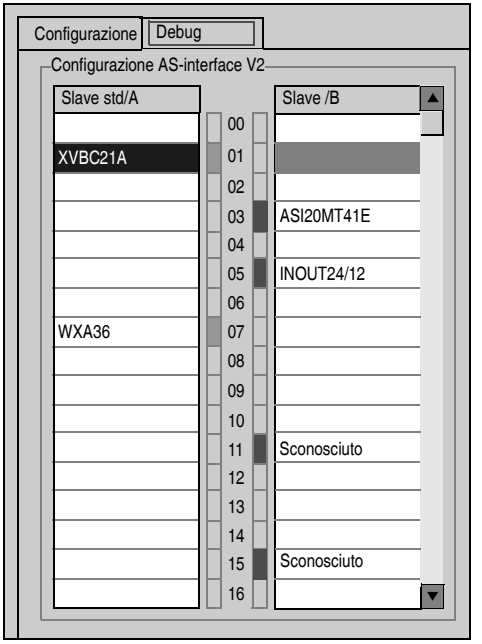
Modifica dell'indirizzo di uno slave

La tabella seguente presenta la procedura necessaria per modificare l'indirizzo di uno slave:

Passo	Descrizione
1	Accedere alla schermata di "Debug".
2	Selezionare uno slave nell'area "Configurazione AS-interface V2".
3	Con il mouse, trascinare verso la cellula corrispondente all'indirizzo desiderato. Illustrazione: trascinare lo slave 3B verso l'indirizzo 15B 

Passo	Descrizione

Risultato:
Il sistema esegue automaticamente un controllo di tutti i parametri dello slave per verificare se l'operazione è possibile.
illustrazione del risultato:



Dopo l'operazione, la diagnostica dello slave all'indirizzo 3B visualizza un messaggio che indica che lo slave non è rilevato, per comunicare che lo slave previsto a questo indirizzo non è più presente. Selezionando l'indirizzo 15B, si ritrovano il profilo e i parametri dello slave che è stato spostato, mentre il nome resta sconosciuto perché la sua presenza in questo indirizzo non è prevista.

Nota: Il profilo e i parametri di uno slave non sono associati al suo nome. Più slave con nomi diversi possono avere gli stessi profili e gli stessi parametri.

Aggiornamento della configurazione del bus AS-Interface in modalità collegato

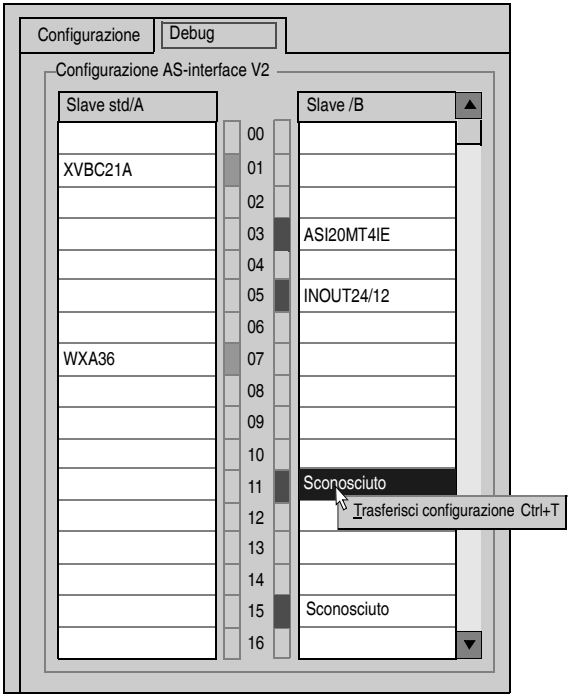
In breve

In modalità collegato, non è possibile eseguire alcuna modifica della schermata di configurazione. È possibile che la configurazione fisica e quella software siano diverse. Qualsiasi differenza di profilo o di parametri di uno slave previsto o meno nella configurazione può essere acquisita nella schermata di configurazione. È infatti possibile trasmettere tutte le modifiche alla schermata di configurazione prima di trasferire la nuova applicazione al controller.
Di seguito viene descritta la procedura per l'acquisizione della configurazione fisica:

Passo	Descrizione
1	Trasferimento della configurazione dello slave desiderato verso la schermata di configurazione.
2	Accettazione della configurazione nella schermata di configurazione.
3	Convalida della nuova configurazione.
4	Trasferimento dell'applicazione al modulo.

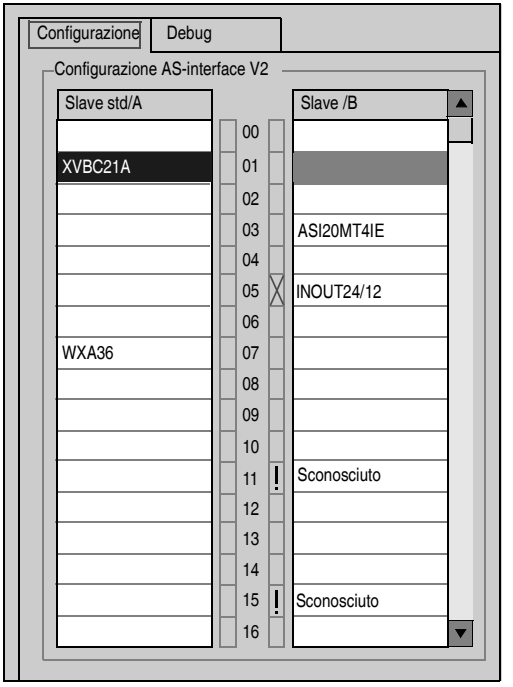
Trasferimento dell'immagine di uno slave verso la configurazione

In caso di rilevamento di uno slave sul bus non previsto dalla configurazione, nell'area "Configurazione AS-interface V2" della schermata di debug, all'indirizzo rilevato viene riportato uno slave "Sconosciuto".
La tabella seguente indica la procedura per il trasferimento dell'immagine dello slave "Sconosciuto" nella schermata di configurazione:

Passo	Descrizione
1	Accedere alla schermata di "Debug".
2	Selezionare lo slave desiderato nell'area "Configurazione AS-interface V2".
3	Fare clic con il pulsante destro del mouse per selezionare "Trasferisci configurazione". Illustrazione:  Risultato: L'immagine dello slave selezionato (immagine del profilo e dei parametri) viene trasferita alla schermata di configurazione.
4	Ripetere l'operazione per ognuno degli slave di cui si desidera trasferire l'immagine verso la schermata di configurazione.

Ritorno alla schermata di configurazione

Una volta tornati alla schermata di configurazione, tutti i nuovi slave (non previsti) trasferiti saranno visibili.
Illustrazione della schermata di configurazione dopo il trasferimento di tutti gli slave:



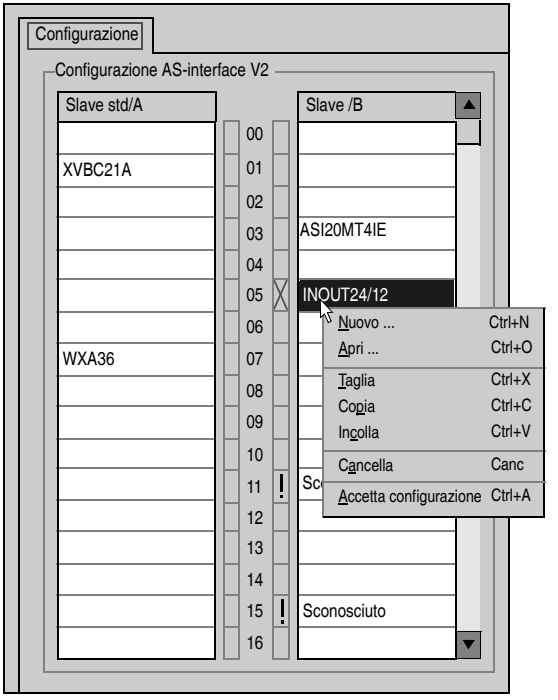
Legenda:

- la croce indica che vi sono differenze tra l'immagine del profilo dello slave trasferito e il profilo desiderato inizialmente nella schermata di configurazione,
- il punto esclamativo significa che nella schermata di configurazione è stato introdotto un nuovo profilo.

Descrizione:
La schermata di configurazione presenta sempre l'immagine permanente della configurazione desiderata (da cui la presenza dello slave in 3B nonostante la variazione d'indirizzo (Vedi *Modifica dell'indirizzo di uno slave, p. 222*)), completata con l'immagine corrente del bus.
I profili e i parametri degli slave previsti che sono visualizzati corrispondono a quelli previsti. I profili e i parametri degli slave sconosciuti che sono visualizzati corrispondono alle relative immagini rilevate.

Procedura per il trasferimento dell'applicazione definitiva verso il modulo

Prima di trasferire una nuova applicazione verso il modulo, per ognuno degli slave l'utente può acquisire l'immagine del profilo e dei parametri che è stata rilevata (cioè trasferita alla schermata di configurazione) oppure modificare la configurazione in modo manuale (Vedi *Procedura di dichiarazione e di configurazione di uno slave, p. 214*).
La tabella seguente descrive la procedura per la convalida e il trasferimento della configurazione definitiva verso il modulo:

Passo	Azione
1	Scollegare in modo software il PC dal modulo. Nota: se il PC è collegato al modulo, dalla schermata di configurazione non è possibile apportare alcuna modifica.
2	Fare clic con il pulsante destro del mouse sullo slave desiderato:
3	Sono possibili due scelte: ● Selezionare "Accetta configurazione" per accettare il profilo **rilevato** dello slave selezionato. Illustrazione:  Per ognuno degli slave contrassegnati con una croce, un messaggio avverte l'utente che questa operazione sostituisce il profilo iniziale, visualizzato sullo schermo, dello slave. ● Selezionare le altre opzioni del menu contestuale per configurare manualmente lo slave selezionato.

Passo	Azione
4	Ripetere l'operazione per ognuno degli slave desiderati della configurazione.
5	Premere "OK" per confermare e creare la nuova applicazione. Risultato: ritorno automatico alla schermata principale.
6	Trasferire l'applicazione al modulo:

Indirizzamento automatico di uno slave AS-Interface V2

In breve

A ogni slave presente sul bus AS-Interface è necessario assegnare, tramite configurazione, un indirizzo fisico univoco. Tale indirizzo deve essere l'immagine di quello dichiarato in TwidoSoft.

Il software TwidoSoft prevede un servizio di indirizzamento automatico degli slave evitando in tal modo l'uso di una console AS-Interface.
Il servizio di indirizzamento automatico consente di:

- sostituire uno slave guasto
- inserire un nuovo slave.

Procedura

La seguente tabella presenta la procedura da seguire per definire il parametro **Indirizzamento automatico**.

Passo	Azione
1	Visualizzare la schermata di configurazione del modulo master AS-Interface V2.
2	Fare clic sulla casella di controllo **Indirizzamento automatico** che si trova nell'area **Modalità master**. **Risultato**: il servizio **Indirizzamento automatico** viene attivato se la casella è selezionata, o sarà inattivo se la casella non è selezionata. **Osservazione**: il parametro **Indirizzamento automatico** è selezionato di default nella schermata di configurazione.

Inserimento di un'apparecchiatura slave in una configurazione AS-Interface V2 esistente

In breve

È possibile inserire un'apparecchiatura in una configurazione AS-Interface V2 esistente senza dover ricorrere a un programmatore tascabile.
L'operazione è possibile nelle seguenti condizioni:

- il servizio **Indirizzamento automatico** della modalità di configurazione è attivo (Vedi *Indirizzamento automatico di uno slave AS-Interface V2, p. 229*),
- nella configurazione fisica manca un solo slave,
- lo slave da inserire è previsto nella schermata di configurazione,
- lo slave dispone del profilo previsto dalla configurazione,
- lo slave ha indirizzo 0(A).

In questo caso, il modulo AS-Interface V2 assegna automaticamente allo slave il valore predefinito nella configurazione.

Procedura

La seguente tabella presenta la procedura necessaria per rendere effettivo l'inserimento automatico di un nuovo slave.

Passo	Azione
1	Aggiungere il nuovo slave nella schermata di configurazione in modalità locale.
2	Eseguire un trasferimento di configurazione verso il controller in modalità collegato.
3	Collegare fisicamente il nuovo slave con indirizzo 0(A) sul bus AS-Interface V2.

Nota: eseguendo l'operazione sopra indicata il numero di volte necessario, è possibile modificare l'applicazione.

Sostituzione automatica di uno slave AS-Interface V2 in errore

Principio

Quando uno slave è dichiarato in errore, è possibile sostituirlo automaticamente con uno slave dello stesso tipo.
La sostituzione avviene senza arrestare il bus AS-Interface V2 e senza interventi particolari, poiché è attivo il servizio di **Indirizzamento automatico** della modalità di configurazione (Vedi *Indirizzamento automatico di uno slave AS-Interface V2, p. 229*).

Sono possibili due casi:

- lo slave sostitutivo è programmato con lo stesso indirizzo tramite un programmatore tascabile ed è dotato dello stesso profilo e sottoprofilo dello slave in errore. Sarà quindi inserito automaticamente nell'elenco degli slave rilevati (LDS) e nell'elenco degli slave attivi (LAS),
- lo slave sostitutivo non è programmato (indirizzo 0(A), slave nuovo) e dispone dello stesso profilo dello slave in errore. In questo caso, prenderà automaticamente l'indirizzo dello slave sostituito e sarà inserito nell'elenco degli slave rilevati (LDS) e nell'elenco degli slave attivi (LAS).

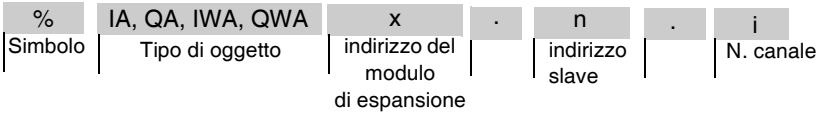
Indirizzamento degli ingressi/uscite associati alle apparecchiature slave collegate sul bus AS-Interface V2

In breve

In questa pagina vengono presentati gli aspetti specifici dell'indirizzamento di ingressi/uscite digitali o analogici delle apparecchiature slave.
Per evitare confusione con gli I/O remoti, vengono introdotti nuovi simboli con sintassi AS-Interface: ad esempio %I**A**.

Illustrazione

Promemoria sul principio di indirizzamento:



Valori specifici

La tabella seguente fornisce i valori specifici per gli oggetti degli slave AS-Interface V2.

Parte	Valori	Commento
IA	-	Immagine dell'ingresso fisico digitale dello slave.
QA	-	Immagine dell'uscita fisica digitale dello slave.
IWA	-	Immagine dell'ingresso fisico analogico dello slave.
QWA	-	Immagine dell'uscita fisica analogica dello slave.
x	1 - 7	Indirizzo del modulo AS-Interface sul bus di espansione
n	da 0A a 31B	L'alloggiamento 0 non è configurabile.
i	0 - 3	-

Esempi

La tabella seguente descrive alcuni esempi di indirizzamento degli I/O.

Oggetto di I/O	Descrizione
%IWA4.1A.0	ingresso analogico 0 dello slave 1A del modulo AS-Interface in quarta posizione sul bus d'espansione.
%QA2.5B.1	uscita digitale 1 dello slave 5B del modulo AS-Interface in seconda posizione sul bus d'espansione.
%IA1.12A.2	ingresso digitale 2 dello slave 12A del modulo AS-Interface in prima posizione sul bus d'espansione.

Scambi impliciti

Gli oggetti sopra descritti vengono scambiati implicitamente, ossia in modo automatico a ogni ciclo del controller.

Programmazione e diagnostica del bus AS-Interface V2

Scambi espliciti

Gli oggetti (parole e bit) associati al bus AS-Interface portano informazioni (ad esempio, funzionamento del bus, stato degli slave ...) e mettono a disposizione comandi supplementari per eseguire la programmazione avanzata della funzione AS-Interface. Questi oggetti vengono scambiati in modo esplicito tra il controller Twido e il master AS-Interface tramite il bus di espansione:

- su domanda del programma utente tramite l'istruzione: ASI_CMD (vedere di seguito "Panoramica sull'istruzione ASI_CMD"),
- tramite la schermata di debug o la tabella di animazione.

Parole di sistema specifiche riservate

Le parole di sistema riservate nel controller Twido per i moduli master AS-Interface consentono di conoscere lo stato della rete: %SW73 è riservata per il primo modulo di espansione AS-Interface e %SW74 per il secondo. Vengono utilizzati solo i primi cinque bit di queste parole e in sola lettura.
La tabella seguente presenta i bit utilizzati:

Parole di sistema	Bit	Descrizione
%SW73 e %SW74	0	stato del sistema (= 1 se la configurazione è corretta, altrimenti 0)
	1	scambio di dati (= 1 se lo scambio dati è attivato, 0 in modalità Data Exchange Off (Vedi *Modalità di funzionamento del modulo interfaccia bus AS-Interface V2, p. 238*))
	2	sistema in stop (= 1 se è attivata la modalità Offline (Vedi *Modalità Offline, p. 238*), altrimenti 0)
	3	istruzione ASI_CMD terminata (= 1 se è terminata, 0 se è in corso)
	4	errore istruzione ASI_CMD (= 1 in caso di errore nell'istruzione, altrimenti 0)

Esempio di utilizzo (per il primo modulo d'espansione AS-Interface):
Prima di utilizzare un'istruzione ASI_CMD, si deve verificare il bit %SW73:X3 per sapere se è in corso un'istruzione: verificare che %SW73:X3 = 1.
Per conoscere l'esito corretto dell'istruzione, verificare che il bit %SW73:X4 sia uguale a 0.

Panoramica sull'istruzione ASI_CMD

L'istruzione ASI_CMD, tramite programma utente, permette di programmare la rete e di ottenere la diagnostica degli slave. I parametri dell'istruzione vengono impostati tramite parole interne (memoria) **%MW**x.
La sintassi dell'istruzione è la seguente:
ASI_CMDn **%MW**x**:**l
legenda:

Simbolo	Descrizione
n	indirizzo del modulo di espansione AS-Interface (da 1 a 7).
x	numero della prima parola interna (memoria) specificato nel parametro (da 0 a 254).
l	lunghezza dell'istruzione in numero di parole (2).

Utilizzo dell'istruzione ASI_CMD

La tabella che segue descrive l'azione dell'istruzione ASI_CMD in base ai valori dei parametri %MW(x) e %MW(x+1) quando necessario. Per le richieste di diagnostica degli slave, il risultato viene restituito in %MW(x+1).

%MWx	%MWx+1	Azione
1	0	esce dalla modalità Offline.
1	1	passa in modalità Offline.
2	0	impedisce lo scambio dei dati tra master e slave (entra in modalità Data Exchange Off).
2	1	autorizza lo scambio dei dati tra master e slave (esce dalla modalità Data Exchange Off).
3	riservato	-
4	Risultato	legge l'elenco degli slave attivi (tabella LAS) dall'indirizzo 0A a 15A (1 bit per slave).
5	Risultato	legge l'elenco degli slave attivi (tabella LAS) dall'indirizzo 16A a 31A (1 bit per slave).
6	Risultato	legge l'elenco degli slave attivi (tabella LAS) dall'indirizzo 0B a 15B (1 bit per slave).
7	Risultato	legge l'elenco degli slave attivi (tabella LAS) dall'indirizzo 16B a 31B (1 bit per slave).
8	Risultato	legge l'elenco degli slave rilevati (tabella LDS) dall'indirizzo 0A a 15A (1 bit per slave).
9	Risultato	legge l'elenco degli slave rilevati (tabella LDS) dall'indirizzo 16A a 31A (1 bit per slave).
10	Risultato	legge l'elenco degli slave rilevati (tabella LDS) dall'indirizzo 0B a 15B (1 bit per slave).
11	Risultato	legge l'elenco degli slave rilevati (tabella LDS) dall'indirizzo 16B a 31B (1 bit per slave).
12	Risultato	legge l'elenco degli errori periferica degli slave (tabella LPF) dall'indirizzo 0A a 15A (1 bit per slave).
13	Risultato	legge l'elenco degli errori periferica degli slave (tabella LPF) dall'indirizzo 16A a 31A (1 bit per slave).
14	Risultato	legge l'elenco degli errori periferica degli slave (tabella LPF) dall'indirizzo 0B a 15B (1 bit per slave).
15	Risultato	legge l'elenco degli errori periferica degli slave (tabella LPF) dall'indirizzo 16B a 31B (1 bit per slave).
16	Risultato	legge lo stato del bus. I dettagli del risultato sono indicati nel paragrafo seguente.

Nota: lo stato del bus viene aggiornato a ogni ciclo del controller. Il risultato dell'istruzione ASI_CMD di lettura del bus è disponibile solo al ciclo successivo del controller.

Informazioni sul risultato dell'istruzione ASI_CMD per la lettura dello stato del bus

In caso di lettura dello stato del bus tramite l'istruzione ASI_CMD (valore del parametro %MWx = 16), il formato del risultato nella parola %MWx+1 è:

%MWx+1		**Descrizione (1=OK, 0=NOK)**
meno significativo	bit 0	Configurazione OK
	bit 1	LDS.0 (slave presente all'indirizzo 0)
	bit 2	Indirizzamento automatico attivo
	bit 3	Indirizzamento automatico disponibile
	bit 4	Modalità di configurazione attiva
	bit 5	Operazione normale attiva
	bit 6	APF (problema di alimentazione)
	bit 7	Offline pronto
più significativo	bit 0	Errore periferica
	bit 1	Scambio di dati attivo
	bit 2	Modalità Offline
	bit 3	Modalità normale (1)
	bit 4	Errore di comunicazione con il master AS-Interface.
	bit 5	Istruzione ASI_CMD in esecuzione
	bit 6	Istruzione ASI_CMD in errore

Informazioni sul risultato dell'istruzione ASI_CMD per la lettura dello stato degli slave

In caso di diagnostica degli slave con l'istruzione ASI_CMD (valore di %MWx compreso tra 4 e 15), lo stato degli slavi viene restituito nei bit (1=OK) della parola %MWx+1. La tabella che segue fornisce le informazioni sul risultato in base al valore della parola %MWx:

%MWx	**%MWx+1**															
valore	byte più significativo								byte meno significativo							
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
4, 8, 12	15A	14A	13A	12A	11A	10A	9A	8A	7A	6A	5A	4A	3A	2A	1A	0A
5, 9, 13	31A	30A	29A	28A	27A	26A	25A	24A	23A	22A	21A	20A	19A	18A	17A	16A
6, 10, 14	15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	0B
7, 11, 15	31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B

Per vedere se lo slave 20B è attivo, occorre eseguire l'istruzione ASI_CMD con la parola interna %MWx di valore 7. Il risultato viene restituito nella parola interna %MWx+1, lo stato dello slave 20B è fornito dal valore del bit 4 del byte meno significativo: se il bit 4 = 1, lo slave 20B è attivo.

Esempi di programmazione dell'istruzione ASI_CMD

Per forzare il passaggio del master AS-Interface (in prima posizione sul bus di espansione) in modalità Offline.

```
LD 1
[%MW0 := 16#0001]
[%MW1 := 16#0001]
LD %SW73:X3          //Se non è in esecuzione alcuna istruzione ASI_CMD, continua
[ASI_CMD1 %MW0:2]    //per forzare il passaggio in modalità Offline
```

Per leggere la tabella degli slave attivi dall'indirizzo 0A a 15A:

```
LD 1
[%MW0 := 16#0004]
[%MW1 := 16#0000          //opzionale]
LD %SW73:X3  //Se non è in esecuzione alcuna istruzione ASI, continua
[ASI_CMD1 %MW0:2]    //per leggere la tabella LAS dall'indirizzo 0A a 15A
```

Modalità di funzionamento del modulo interfaccia bus AS-Interface V2

In breve

Il modulo interfaccia AS-Interface TWDNOI10M3 dispone di tre modalità di funzionamento, ciascuna delle quali risponde a particolari esigenze. Tali modalità sono:

- la modalità protetta,
- la modalità Offline,
- la modalità Data Exchange Off.

L'utilizzo dell'istruzione ASI_CMD (Vedi *Panoramica sull'istruzione ASI_CMD, p. 234*) in un programma utente consente di entrare o uscire da queste modalità.

Modalità protetta

La modalità di funzionamento protetto viene generalmente utilizzata per un'applicazione in funzionamento. A questo scopo, è necessario che il modulo AS-Interface V2 sia configurato in TwidoSoft. Quest'ultimo:

- verifica continuamente che l'elenco degli slave rilevati corrisponda all'elenco degli slave previsti,
- sorveglia l'alimentazione.

In questa modalità, uno slave viene attivato solo se è stato dichiarato nella configurazione e rilevato.
All'avvio o durante la configurazione, il controller Twido forza il modulo AS-Interface in modalità protetta.

Modalità Offline

Non appena in modalità Offline, il modulo effettua dapprima un azzeramento di tutti gli slave presenti e arresta gli scambi sul bus. Durante la modalità Offline, le uscite vengono forzate a zero.
Oltre all'impiego del pulsante PB2 sul modulo AS-Interface TWDNOI10M3, è possibile accedere alla modalità Offline via software tramite l'istruzione ASI_CMD (Vedi *Esempi di programmazione dell'istruzione ASI_CMD, p. 237*), e analogamente per uscire dalla modalità e tornare in modalità protetta.

Modalità Data Exchange Off

Con l'attivazione della modalità Data Exchange Off, gli scambi sul bus continuano a funzionare ma i dati non vengono più aggiornati.
Questa modalità è accessibile solo tramite l'istruzione ASI_CMD (Vedi *Utilizzo dell'istruzione ASI_CMD, p. 235*).

Installazione e configurazione del bus di campo CANopen

10

In breve

Contenuto del capitolo

Questo capitolo descrive come installare e configurare il modulo master CANopen TWDNCO1M e i dispositivi slave sul bus di campo CANopen.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
10.1	Panoramica del bus di campo CANopen	241
10.2	Implementazione del bus CANopen	255

10.1 Panoramica del bus di campo CANopen

In breve

Contenuto della sezione

Questa sezione fornisce informazioni generali sulla tecnologia del bus di campo CANopen e presenta la terminologia specifica CAN che verrà utilizzata nel resto di questo capitolo.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Elementi di base CANopen	242
Informazioni su CANopen	243
CANOpen Boot-Up	246
Process Data Object (PDO) - Trasmissione	249
Accesso ai dati tramite scambi espliciti (SDO)	251
"Node Guarding e Life Guarding	252
Gestione del bus interno	254

Elementi di base CANopen

Introduzione

Di seguito sono riportate le spiegazioni di alcuni termini tecnici e acronimi utili per comprendere i concetti di base della comunicazione sulla rete CANopen.

File EDS

EDS (Electronic Data Sheet)
Un file EDS descrive le proprietà di comunicazione di un dispositivo sulla rete CAN (velocità, tipi di trasmissione, offerta di I/O e così via). Esso viene fornito dal produttore del dispositivo ed è utilizzato dallo strumento di configurazione per configurare un nodo, in modo simile ai driver del sistema operativo.

PDO

PDO (Process Data Object)
Pacchetto dati CANopen contenente dati di I/O.
È possibile distinguere tra:
- Transmit-PDOs (TPDO con dati forniti da un nodo) e
- Receive PDOs (RPDOcon dati che verranno utilizzati da un nodo.

La direzione di trasmissione viene sempre considerata dal punto di vista di un nodo. Un PDO non contiene necessariamente l'immagine intera dei dati di un nodo, sia per TPDO sia per RPDO. Normalmente, i dati degli ingressi analogici e i dati degli ingressi digitali sono divisi in TPDO diversi e lo stesso accade per le uscite.

SDO

SDO (Service Data Object)
Pacchetto dati CANopen contenente parametri.
In generale, gli SDO vengono utilizzati per leggere parametri da o per scrivere parametri su unità mentre l'applicazione è in esecuzione.

COB-ID

COB-ID (Communication Object Identifier)
Ogni pacchetto dati CANopen inizia con un COB-ID, che ha la funzione di identificatore del pacchetto stesso. Durante la fase di configurazione, ogni nodo riceve i COB-ID per i pacchetti di cui è provider o consumer.

Informazioni su CANopen

Introduzione

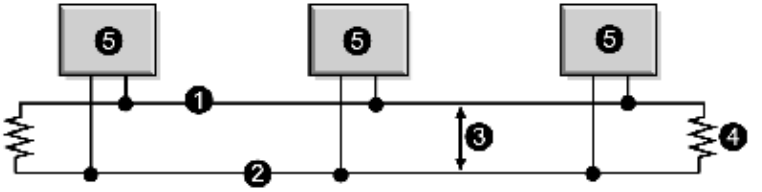
CANopen è un protocollo di bus di campo standard per sistemi di controllo industriale. In particolare, è idoneo per i controller in tempo reale, perché rappresenta una soluzione efficace e conveniente per le applicazioni industriali integrate e trasportabili.

Protocollo CANopen

Il protocollo CANopen è stato creato come subset di CAL. Grazie alla definizione dei profili, è possibile adattarlo in modo molto specifico per l'uso con componenti industriali standard. CANopen è uno standard CiA (CAN in Automation) che ha subito trovato larga diffusione sul mercato. In Europa, attualmente CANopen è riconosciuto come lo standard di settore per i sistemi industriali basati su CAN.

Livello fisico

CAN utilizza una linea bus differenziale a due fili (ritorno comune). Il segnale CAN è la differenza tra i livelli di tensione dei fili CAN-high e CAN-low. Vedere la figura seguente.
Il diagramma seguente mostra i componenti del livello fisico di un bus CAN a due fili:



1 filo CAN-high

2 filo CAN-low

3 differenza di potenziale tra i segnali CAN-high e CAN-low

4 connettore resistenza 120Ω

5 nodo

È possibile instradare in parallelo i fili del bus, intrecciarli o schermarli secondo quanto richiesto dalla compatibilità elettromagnetica. Una struttura a linea singola riduce al minimo la riflessione.

Profili CANopen

Profilo di comunicazione

La famiglia dei profili CANopen si basa su un "profilo di comunicazione" che specifica i principali meccanismi di comunicazione e le relative descrizioni (DS301).

Profilo dell'apparecchiatura

Alla sezione "Profili delle apparecchiature" sono descritti i tipi di apparecchiature più importanti utilizzati nell'automazione industriale, oltre alla definizione delle funzionalità delle apparecchiature stesse.

Tra gli esempi delle apparecchiature standard descritte vi sono:

- moduli con ingressi e uscite digitali e analogici (DS401),
- motori (DS402),
- apparecchiature di controllo (DSP403),
- controller a loop chiuso (DSP404),
- PLC (DS405),
- encoder (DS406).

Configurazione delle apparecchiature tramite il bus CAN

La possibilità di configurare le apparecchiature tramite il bus CAN è uno dei principi di base dell'autonomia richiesta dai produttori per ogni famiglia di profilo.

Specifiche generali dei profili CANopen

CANopen è costituito da una serie di profili per i sistemi CAN con le seguenti specifiche:

- sistema a bus aperto,
- scambio di dati in tempo reale senza sovraccarico del protocollo,
- design modulare con possibilità di ridimensionamento,
- interoperabilità e intercambiabilità delle apparecchiature,
- supporto da parte di molti produttori internazionali,
- configurazione di rete standardizzata,
- accesso a tutti i parametri delle apparecchiature,
- sincronizzazione e circolazione dei dati ciclici di processo e/o dei dati determinati da eventi e possibilità di tempi di risposta brevi del sistema.

Certificazione di prodotto CANopen

Tutti i produttori che offrono al mercato prodotti certificati CANopen appartengono all'associazione CiA. In qualità di membro attivo dell'associazione CiA, Schneider Electric Industries SAS sviluppa i propri prodotti seguendo le raccomandazioni di standardizzazione che essa emette.

Standard CAN

Le specifiche CANopen sono definite dall'associazione CiA e sono consultabili, con alcune limitazioni, sul sito del gruppo all'indirizzo http://www.can-cia.com. Molti produttori mettono a disposizione i codici sorgenti per le apparecchiature master e slave.

Nota: per ulteriori informazioni sulle specifiche e sui meccanismi standard CANopen, visitare il sito CiA all'indirizzo http://www.can-cia.de/.

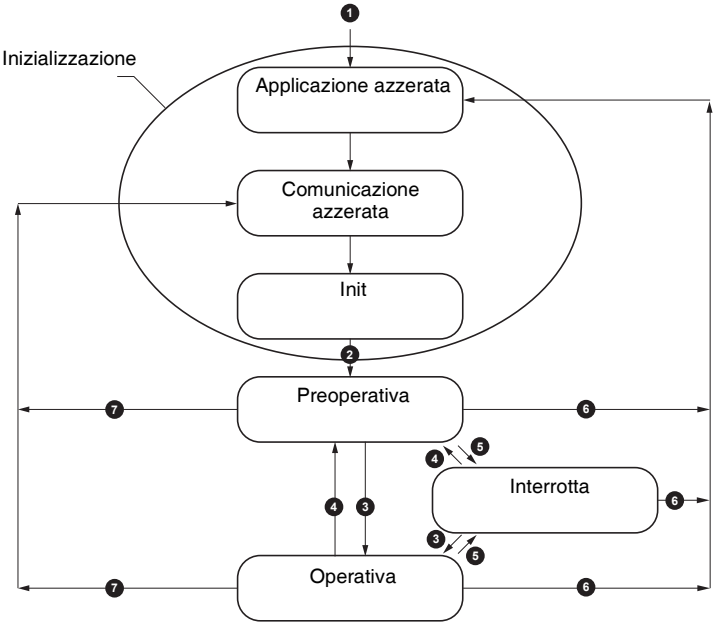
Comunicazione su una rete CANopen

Il profilo di comunicazione si basa su funzionalità e protocolli CAL.
Esso mette a disposizione dell'utente due tipi di scambio: SDO e PDO.
All'accensione, l'apparecchiatura entra in fase di inizializzazione e quindi in stato preoperativo. A questo punto, è autorizzata solo la comunicazione SDO. Dopo la ricezione di un comando di avvio, l'apparecchiatura passa allo stato operativo. È quindi possibile eseguire scambi PDO e le comunicazioni SDO restano possibili.

CANOpen Boot-Up

Boot-up Procedura

La configurazione minima del dispositivo specifica una procedura di avvio abbreviata. Questa procedura è illustrata nel seguente diagramma:



Legenda

Numero	Descrizione
1	Accensione del modulo
2	Dopo l'inizializzazione, il modulo passa automaticamente allo stato PRE-OPERATIONAL.
3	Indicazione della funzionalità NMT: START REMOTE NODE
4	Indicazione della funzionalità NMT: PRE-OPERATIONAL
5	Indicazione della funzionalità NMT: STOP REMOTE NODE
6	Indicazione della funzionalità NMT: RESET NODE
7	Indicazione della funzionalità NMT: RESET COMMUNICATION

Oggetti CANopen attivi che dipendono dalla macchina di stato

Le X riportate nella tabella seguente indicano quali sono gli oggetti CANopen attivi per quali stati della macchina di stato.

	Inizializzazione	Preoperativa	Operativa	Applicazione interrotta
Oggetto PDO			X	
Oggetto SDO		X	X	
Emergenza		X	X	
Avvio	X		X	
NMT		X	X	X

Reset Application

Il dispositivo passa allo stato "Reset Application":
- dopo l'avvio del dispositivo,
- oppure tramite la funzionalità "Reset Node" della gestione di rete (NMT).

In questo stato, il profilo del dispositivo viene inizializzato e tutte le informazioni sul relativo profilo vengono riportate ai valori predefiniti. Al termine dell'inizializzazione, il dispositivo passa automaticamente allo stato "Reset Communication".

Reset Communication

Il dispositivo passa allo stato "Reset Communication":
- dopo lo stato "Reset Application",
- oppure tramite la funzionalità "Reset Communication" della gestione di rete (NMT).

In questo stato, tutti i parametri (valore standard, a seconda della configurazione del dispositivo) degli oggetti di comunicazione supportati (oggetti riguardanti l'identificazione dei dispositivi quali ad esempio tipo di dispositivo, heartbeat e così via: 1000H - 1FFFH) sono salvati nella directory oggetto. Il dispositivo passa quindi in modo automatico nello stato "Init".

Init

Il dispositivo passa alla modalità "Init" dopo aver assunto lo stato "Reset Communication".
Questo stato consente di:
- definire gli oggetti di comunicazione richiesti (SDO, PDO, Emergency),
- installare le corrispondenti funzionalità CAL,
- configurare il CAN-Controller.

L'inizializzazione del dispositivo è completa e questa passa automaticamente allo stato "Pre-Operational".

Nota: Il modulo master CANopen TWDNCO1M non supporta la modalità SYNC.

Pre-Operational

Il dispositivo passa allo stato "Pre-Operational":
- dopo lo stato "Init",
- alla ricezione dell'indicazione NMT "Enter Pre-Operational" se si trovava nello stato Operational.

Quando il dispositivo si trova in questo stato, è possibile modificarne la configurazione. Per leggere o scrivere i dati riguardanti il dispositivo, tuttavia, è possibile utilizzare solo SDO.
Al termine della configurazione, il dispositivo passa in uno dei seguenti stati alla ricezione dell'indicazione corrispondente:
- "Stopped" alla ricezione dell'indicazione NMT "STOP REMOTE NODE",
- "Operational" alla ricezione dell'indicazione NMT "START REMOTE NODE".

Stopped

Il dispositivo passa allo stato "Stopped" alla ricezione dell'indicazione "Node stop" (funzionalità NMT) se si trovava nello stato "Pre-Operational" oppure "Operational". In questo stato, non è possibile configurare il dispositivo. Non è disponibile alcuna funzionalità per la lettura e la scrittura dei dati relativi al dispositivo (SDO). Resta attiva solo la funzione di monitoraggio degli slave ("Node guarding").

Operational

Il dispositivo passa allo stato "Operational" se si trovava nello stato "Pre-Operational" alla ricezione dell'indicazione "Start Remote Node".
Quando si inizia la rete CANopen utilizzando le funzionalità NMT "Node start" nello stato "Operational", è possibile utilizzare tutte le funzionalità del dispositivo. Le comunicazioni possono utilizzare PDO e SDO.

Nota: modifiche alla configurazione in modalità "Operativa" possono portare a conseguenze impreviste. Per questo motivo, è necessario eseguirle solo in modalità "preoperativa".

Process Data Object (PDO) - Trasmissione

Definizione di PDO

I PDO sono oggetti che forniscono l'interfaccia di comunicazione con i dati di processo, permettendone lo scambio in tempo reale. Il set di PDO di un dispositivo CANOpen descrive gli scambi impliciti sulla rete tra il dispositivo in questione e i suoi partner di comunicazione.
Lo scambio dei PDO è autorizzato quando il dispositivo è in modalità "Operational".

Tipi di PDO

Sono disponibili due tipi di PDO:
- PDO trasmessi dal dispositivo, spesso denominati:Transmit PDO o Tx-PDO o TPDO,
- PDO ricevuti dal dispositivo, spesso denominati:Receive PDO o Rx-PDO o RPDO.

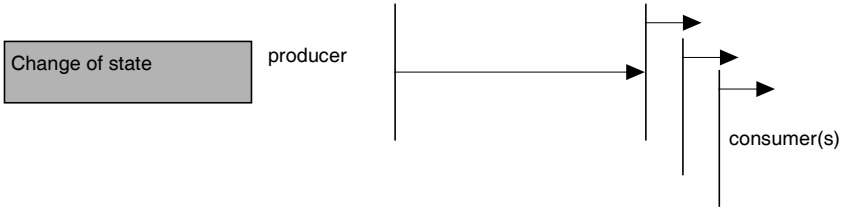
PDO Producers and Consumers

I PDO si basano su un modello "Producer / Consumer". Il dispositivo che emette un PDO è denominato "producer" mentre quello che lo riceve è chiamato "consumer". In tal modo, quando si scrive un'uscita sul modulo master TWDNCO1M si invia un TPDO associato al master che contiene il valore dell'uscita da aggiornare. In questo caso, il master è "producer" del PDO mentre lo slave ne è il "consumer". Analogamente, quando si aggiorna un'uscita con la trasmissione di un RPDO da parte del modulo master, esso è il "consumer".

Modalità di trasmissione dei PDO

Oltre ai dati da trasportare, è possibile configurare il tipo di scambio per ogni PDO. Il modulo master TWDNCO1M può scambiare PDO secondo le seguenti modalità di trasmissione:

Numero di modalità	Tipo di modalità	Nome della modalità
254 o 255	Asincrona	Change of state



Change of state (Modalità 254 e 255)

"Change of state" corrisponde alla modifica di un valore d'ingresso o controllo degli eventi. Immediatamente dopo la modifica, i dati sono inviati sul bus. Il controllo degli eventi permette di ottimizzare l'uso della larghezza di banda del bus, perché si trasmettono solo le variazioni e non tutta l'immagine del processo. In tal modo, è possibile ottenere tempi di risposta molto ridotti, perché quando il valore d'ingresso subisce una modifica non è necessario attendere la richiesta successiva del master. Quando si seleziona "change of state" per la trasmissione dei PDO, occorre tenere presente che è probabile che si verifichino numerosi eventi in contemporanea. Ciò può provocare dei ritardi, dovuti all'attesa della trasmissione sul bus degli PDO con priorità inferiore. Occorre inoltre evitare le situazioni in cui la continua modifica di ingressi con PDO di elevata priorità possa bloccare il bus, fenomeno denominato anche "babbling idiot".

Nota: In generale, è consigliabile utilizzare la trasmissione dei PDO con moduli di ingresso analogico solo se sono impostati la modalità Delta (oggetto 6426H) o il tempo di inibizione (oggetti da 1800H a 1804H, sottoindice 3), in modo da evitare un sovraccarico del bus.

Accesso ai dati tramite scambi espliciti (SDO)

Che cos'è uno SDO

Service Data Objects (SDO) consentono di accedere ai dati dei dispositivi tramite richieste esplicite.
Il servizio SDO è disponibile quando il dispositivo si trova in stato "Operational" oppure "Pre-Operational".

Tipi di SDO

Sono disponibili due tipi di SDO:
- SDO di lettura (SDO di scaricamento),
- SDO di scrittura (SDO di caricamento).

Modello client/ server

Il protocollo SDO si basa su un modello "client/server".
Per uno SDO di scaricamento
Il client invia una richiesta che indica l'oggetto da leggere.
Il server restituisce i dati contenuti nell'oggetto.
Per uno SDO di caricamento
Il client invia una richiesta che indica l'oggetto da scrivere e il valore desiderato.
Dopo l'aggiornamento dell'oggetto, il server restituisce un messaggio di conferma.
Per un SDO non elaborato
In entrambi i casi, se non è stato possibile elaborare uno SDO il server restituisce un codice di errore (abort code).

"Node Guarding e Life Guarding

Definizione di Life-Time

Il parametro Life time è calcolato come segue:
Life Time = Guard Time x Life Time Factor
L'oggetto 100CH contiene il parametro 'Guard Time espresso in millisecondi.
L'oggetto 100DH contiene il parametro Life Time Factor.

Attivazione del monitoraggio

Se uno di questi due parametri è impostato a "0" (configurazione predefinita) il modulo non esegue il monitoraggio (nessun "Life Guarding").
Per attivare il monitoraggio, è necessario immettere almeno il valore 1 nell'oggetto 100DH e specificare un tempo in millisecondi nell'oggetto 100CH.

Garanzia di affidabilità del funzionamento

Per garantire un funzionamento affidabile, è consigliabile immettere un valore di "Life time factor" pari a 2.
In caso contrario, se dovesse verificarsi un ritardo, ad esempio dovuto all'elaborazione di messaggi di priorità più elevata o all'elaborazione sul master di "Node Guarding", il modulo passa allo stato "Pre-Operational" senza generare alcun errore.

Importanza del monitoraggio

Questi due meccanismi di monitoraggio sono particolarmente importanti per il sistema CANopen, poiché in generale i dispositivi non funzionano in modalità di evento controllato.

Monitoraggio degli slave

Il monitoraggio si svolge come segue:

Fase	Descrizione
1	Il master imposta "Remote Frames" (remote transmit requests) sugli "Guarding COB-IDs" degli slave da monitorare.
2	Gli slave interessati rispondono inviando il messaggio di "Guarding". Tale messaggio contiene lo "Status Code" dello slave e il "Toggle Bit" che deve cambiare dopo ogni messaggio.
3	Il master confronta le informazioni di "Status" e di "Toggle Bit": Se non si trovano nello stato previsto dal master NMT o se non riceve alcuna risposta, il master considera che si sia verificato un errore nello slave.

Monitoraggio del master

Se il master richiede i messaggi "Guarding" con frequenza strettamente ciclica, lo slave può rilevare eventuali guasti che interessano il master.
Se lo slave non riceve una richiesta dal master entro l'intervallo di "Life Time" definito (errore di Guarding), considera che si sia verificato un guasto nel master (funzione "Watchdog").
In tal caso, le uscite corrispondenti passano allo stato di errore e lo slave ritorna alla modalità "Pre-Operational".

Nota: la richiesta "Remota" del master ottiene una risposta anche se non si sono immessi valori per gli oggetti "Guard Time" e "Life Time Factor". Il monitoraggio a tempo viene attivato solo quando i valori dei due oggetti sono maggiori di 0. Valori tipici per il parametro "Guard Time" sono compresi tra 250 ms e 2 secondi.

"Guarding" Protocollo

Il valore del "Toggle Bit" (t) inviato nel primo messaggio "Guarding" è "0".
Il bit cambia ("toggles") quindi a ogni messaggio successivo di "Guarding", permettendo così di rilevare l'eventuale perdita di messaggi.
L'estremità del bus indica lo stato della rete (s) nei sette bit rimanenti:

Stato di rete	Risposta
Stopped	0x04 o 0x84
Pre-operational	0x7F o 0xFF
Operational	0x05 o 0x85

Gestione del bus interno

Passaggio del bus interno allo stato di "Stop

Il bus interno passa automaticamente dallo stato di "Stop" allo stato di "Run" quando il modulo di comunicazione passa dallo stato "Pre-operational" allo stato "Operational".
Quando il bus interno passa allo stato "Stop", tutte le uscite del modulo di espansione sono impostate a zero.
Le uscite del modulo di comunicazione sono mantenute nello stato corrente.

Configurazione dei moduli di espansione

Il bus interno consente di aggiornare la configurazione dei parametri dei moduli di espansione digitali e analogici.
I parametri sono inviati al modulo di comunicazione quando il bus si trova nello stato "Stop".
I nuovi parametri di configurazione sono riconosciuti quando il bus passa in stato "Run".

10.2 Implementazione del bus CANopen

Panoramica

Introduzione

Questa sezione descrive come implementare il bus di campo CANopen sul sistema del controller Twido tramite il modulo master CANopen TWDNCO1M.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Panoramica	256
Configurazione hardware	257
Metodologia di configurazione	258
Dichiarazione del master CANopen	260
Dichiarazione sulla rete di slave CANopen	261
Mappatura oggetti CANopen	265
Collegamento oggetti CANopen	269
Rappresentazione simbolica degli oggetti CANopen	271
Indirizzamento dei PDO del master CANopen	273
Programmazione e diagnostica del bus di campo CANopen	274

Panoramica

Requisiti hardware e software

Per implementare il bus CANopen sul sistema del controller Twido, sono necessari il seguente hardware e software:

Hardware	Requisiti
Controller di base modulare o compatto Twido	**Base compatta:** • TWDLC•24DRF • TWDLCA•40DRF **Base modulare:** • TWDLMDA20••• • TWDLMDA40•••
Master CANopen	1 modulo master CANopen: TWDNCO1M
Dispositivi slave CANopen	Massimo 16 slave CANopen
Connettori e cavi CANopen	
Cavo di programmazione per il controller Twido	

Software	Requisiti
Software di configurazione del controller Twido	TwidoSoft V3.0 o successiva

Procedura di implementazione CANopen

La procedura seguente descrive come installare, configurare e utilizzare la rete CANopen:

Passo	Descrizione
1	Configurazione hardware
2	Metodologia di configurazione
3	Dichiarazione del master CANopen
4	Dichiarazione sulla rete di slave CANopen
5	Mappatura oggetti CANopen
6	Collegamento oggetti CANopen
7	Rappresentazione simbolica oggetti CANopen
8	Diagnostica di rete CANopen
Le sottosezioni seguenti forniscono una descrizione dettagliata di ogni passo della procedura.	

Configurazione hardware

Installazione del modulo master TWDNCO1M

Installare il modulo master TWDNCO1M su un sistema controller Twido (montaggio su guida DIN oppure a pannello) e collegarlo al bus interno del controller Twido (per ulteriori dettagli, consultare TwdoHW - Installazione di un modulo di espansione). Procedere come segue:

Passo	Azione	Descrizione
1	Preparazione dell'installazione	Consultare il *manuale di riferimento hardware dei controller programmabili Twido (TWD USE 10AE)* per istruzioni su: ● posizioni di montaggio corrette per i moduli Twido, ● aggiunta e rimozione dei componenti Twido da una guida DIN, ● montaggio diretto sulla superficie di un pannello, ● spazio minimo intorno ai moduli in un pannello di controllo.
2	Montaggio del modulo TWDNCO1M	Installare il modulo master TWDNCO1M su una guida DIN o su un pannello. Per ulteriori informazioni, consultare TwdoHW - Installazione di un modulo di espansione.
3	Collegamento del modulo al bus del controller Twido	Collegare il modulo master CANopen al bus interno del controller Twido (per ulteriori informazioni, consultare TwdoHW - Installazione di un modulo di espansione).
4	Cablaggio e raccordi CANopen	Per collegare l'alimentazione e le linee di segnale del bus CANopen, seguire le istruzioni di cablaggio e di collegamento descritte in Cablaggio e raccordi CANopen.

Metodologia di configurazione

Panoramica

La configurazione CANopen è eseguita tramite lo strumento di configurazione CANopen, disponibile su TwidoSoft V3.0 o successiva.

Nota:
1. La configurazione di rete, master e slave CANopen, oltre alla configurazione dei parametri di comunicazione avviene esclusivamente in modalità non in linea.
2. In modalità in linea, non è possibile alcuna modifica della configurazione CANopen.
3. In modalità in linea, è possibile regolare alcuni parametri, ad esempio i parametri di indirizzamento PDO %IWC e %QWC.

Metodologia di configurazione

La seguente tabella descrive le diverse fasi dell'implementazione software del bus CANopen.

Modalità	Fase	Descrizione
Non in linea	Dichiarazione del modulo TWDNCO1M	Selezionare un numero di alloggiamento disponibile per installare il modulo master TWDNCO1M sul bus di espansione Twido.
	Configurazione della rete CANopen	La configurazione della rete CANopen si basa sulle seguenti operazioni: ● importazione dei file EDS di tutti i dispositivi slave del catalogo di rete, ● aggiunta dei dispositivi slave dal catalogo della rete CANopen.
	Mappatura dei PDO	Eseguire la mappatura degli oggetti TPDO e RPDO per ogni dispositivo slave dichiarato sulla rete.
	Collegamento dei PDO	Collegare ogni PDO slave al corrispondente PDO del modulo master.
Locale o collegato	Simbolizzazione (opzionale)	Rappresentazione simbolica delle variabili associate ai dispositivi slave.
	Programmazione	Programmazione della funzione CANopen.
In linea	Trasferimento	Trasferimento dell'applicazione nel controller.
	Debug	Debug dell'applicazione tramite: ● La schermata di debug che permette da una parte la visualizzazione degli slave (indirizzo, parametri) e dall'altra l'assegnazione degli slave agli indirizzi desiderati. ● Schermate di diagnostica che consentono di identificare gli errori.

Nota: la dichiarazione e l'eliminazione del modulo master CANopen TWDNCO1M sul bus di espansione è uguale a qualsiasi altro modulo di espansione. Sul bus di espansione Twido, tuttavia, è consentito un solo modulo master CANopen. Il programma di interfaccia utente TwidoSoft non consente di aggiungere altri moduli CANopen.

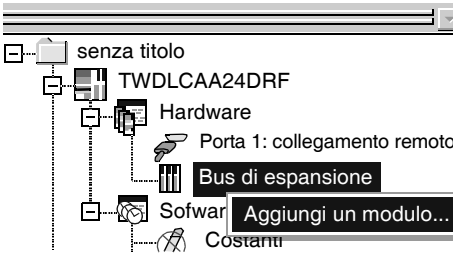
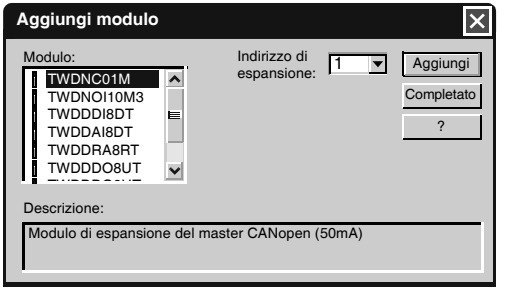
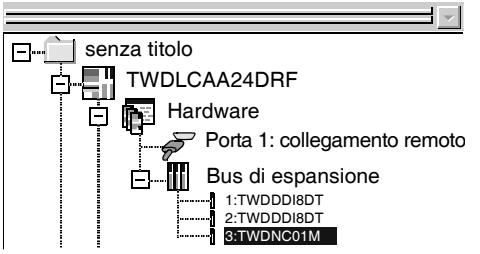
Precauzioni prima della connessione

Prima di collegare in modo software il PC al controller e per evitare eventuali problemi di rilevamento:

- Verificare che non vi siano slave presenti fisicamente sul bus all'indirizzo 127. 127 è un indirizzo riservato e impostato dal produttore assegnato al modulo master TWDNCO1M.
- Verificare che sul bus CANopen non siano installati slave con indirizzi duplicati.

Dichiarazione del master CANopen

Procedura La tabella seguente mostra i vari passaggi della dichiarazione del master CANopen.

Passo	Azione	Commento
1	Nel browser dell'applicazione TwidoSoft, fare clic con il pulsante destro del mouse su **Bus di espansione** → **Aggiungi un modulo...**	
2	Quando viene visualizzata la finestra di dialogo **Aggiungi un modulo**: • Selezionare **TWDNCO1M.** • Fare clic su **Aggiungi.** • A questo punto, è possibile continuare ad aggiungere altri moduli di espansione (massimo 7) da includere nel sistema Twido. **Nota:** è ammesso un solo modulo master CANopen TWDNCO1M. • Fare clic su **Completato.**	Sono supportati solo i controller TWDC•A24DRF, TWDCA•40DRF, TWDLMDA20••• e TWDLMDA40•••. 
3	Viene visualizzata una struttura di espansione del bus simile all'esempio seguente. **Nota:** sul bus Twido è possibile inserire un master TWDNCO1M in qualsiasi alloggiamento di espansione numerato da 1 a 7.	

Dichiarazione sulla rete di slave CANopen

Panoramica

La dichiarazione sulla rete degli slave CANopen si svolge in tre fasi:

1. importazione dei file EDS delle apparecchiature slave CANopen nel catalogo del configuratore CANopen di Twido,
2. creazione della rete CANopen mediante l'aggiunta di massimo 16 apparecchiature slave dal catalogo della rete,
3. configurazione dei parametri di gestione della rete, cioè dei parametri relativi alla velocità di rete e al protocollo di controllo degli errori.

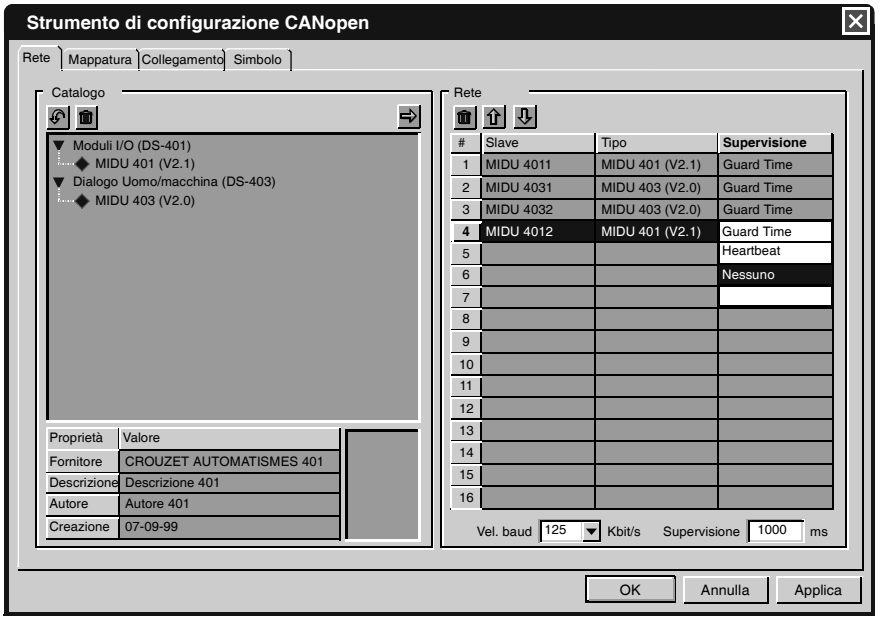
Configuratore CANopen

Dal browser dell'applicazione di TwidoSoft, fare clic con il pulsante destro del mouse sul nome del modulo master e selezionare **Hardware** → **Bus di espansione** → **TWDNCO1M** → **Configura**

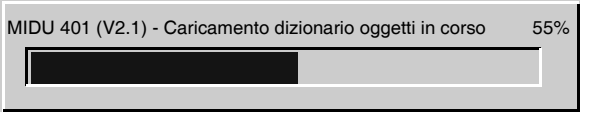
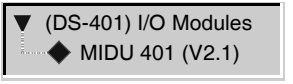
Risultato: viene visualizzato lo strumento di configurazione CANopen, come mostrato nella seguente sottosezione.

Finestra di dialogo Rete

La configurazione della rete CANopen e la dichiarazione degli slave avviene tramite la finestra di dialogo **Rete** del configuratore CANopen di TwidosSoft, riportata di seguito:



Importazione dei profili degli slave La tabella seguente descrive come si esegue l'importazione dei profili degli slave CANopen (file .EDS) nel catalogo dello strumento di configurazione CANopen:

Passo	Azione
1	Dall'area **Catalogo** della finestra di dialogo Rete, fare clic sull'icona **Importa**.  **Risultato:** viene visualizzata la finestra di dialogo Apri del sistema operativo.
2	Sfogliare fino ad aprire la cartella contenente i file **EDS** delle apparecchiature slave CANopen da aggiungere al catalogo. **Risultato:** nella finestra di dialogo Apri vengono visualizzati i file EDS disponibili:
3	Selezionare un file EDS **("nomefile".EDS)** dall'elenco e fare clic su **Apri.** **Risultato:** lo strumento di configurazione CANopen carica il dizionario degli oggetti per l'apparecchiatura selezionata. **Nota:** questo processo può richiedere anche diversi minuti, a seconda delle dimensioni del file EDS selezionato. La barra di avanzamento indica lo stato di completamento del processo di caricamento, come mostrato nell'esempio seguente: 
4	Attendere fino al termine del caricamento e ripetere i passi 2 e 3 per tutti i nuovi profili di slave da aggiungere al catalogo. **Nota:** questo processo deve essere eseguito solo una volta, perché tutti i profili di apparecchiatura e i dizionari degli oggetti elencati nel catalogo caricato vengono memorizzati in TwidoSoft.
5	Per visualizzare le proprietà apparecchiatura di uno slave CANopen: **1.** Fare clic due volte sul tipo di apparecchiatura elencato nel catalogo. **Esempio:**  . **Risultato:**  . **2.** fare clic una volta sul profilo dello slave (ad esempio, MIDU 401 V2.1). **Risultato:** le proprietà apparecchiatura dello slave CANopen selezionato sono visualizzate nella metà inferiore dell'area del Catalogo che riporta: • il nome del produttore, ad esempio Crouzet Automatismes 401, • il profilo dello slave, ad esempio Descrizione 401, • il nome dell'autore, ad esempio Autore 401, • la data di creazione del profilo, ad esempio 07-09-99.
6	Per eliminare un profilo slave dal Catalogo, selezionare il nome dell'apparecchiatura nella relativa finestra e fare clic sull'icona **Elimina** .  **Nota:** nel catalogo di rete CANopen è possibile memorizzare più profili di apparecchiature di quanti siano effettivamente necessari per la configurazione del bus CANopen corrente. I profili già caricati nel catalogo possono essere utilizzati successivamente.
7	Premere il pulsante **Applica** per confermare le modifiche al catalogo e salvare i profili degli slave nel progetto TwidoSoft.

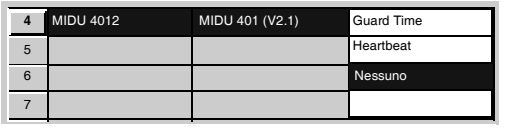
Creazione della rete CANopen

La tabella seguente descrive come è possibile dichiarare le apparecchiature slave sulla rete CANopen di Twido. Si tenga presente che è possibile dichiarare solo gli slave i cui profili EDS sono già stati aggiunti o sono già memorizzati nel catalogo.

Passo	Azione
1	Nell'area **Catalogo** della finestra di dialogo Rete, selezionare il profilo slave dall'elenco delle apparecchiature disponibili memorizzate. **Risultato:** L'icona **Aggiungi** appare  nell'angolo in alto a destra della struttura del catalogo.
2	Fare clic una volta sull'icona **Aggiungi** .  **Risultato:** l'apparecchiatura slave viene aggiunta alla tabella degli slave di rete. **Note:** ● Sulla rete CANopen Twido è possibile dichiarare al massimo 16 slave. ● L'apparecchiatura slave appena dichiarata assume l'indirizzo di nodo con il primo indice disponibile. Se ad esempio le apparecchiature slave sono dichiarate agli indirizzi di nodo 1, 2 e 4, la nuova apparecchiatura assumerà in modo predefinito l'indirizzo 3.
3	È possibile assegnare l'apparecchiatura slave a qualsiasi indirizzo di nodo disponibile, da 1 a 16. Per spostare un'apparecchiatura slave all'indirizzo di nodo desiderato, utilizzare le icone delle frecce di spostamento **su e** giù .
4	Ripetere i passi da 1 a 3 per ogni nuova apparecchiatura slave da dichiarare sulla rete CANopen.
5	Per eliminare un'apparecchiatura slave dalla rete, selezionare il nome dell'apparecchiatura nella tabella degli slave e fare clic sull'icona **Elimina** . 
6	Premere il pulsante **Applica** per confermare le modifiche e salvare la configurazione di rete nel progetto TwidoSoft.

Configurazione dei parametri di gestione della rete

La procedura seguente descrive come eseguire la configurazione dei parametri di gestione della rete quali Baudrate o velocità di rete, life-time e protocollo di controllo degli errori.

Passo	Azione
1	Nella finestra di dialogo Rete, selezionare il Baudrate (velocità di rete) dall'elenco a discesa: **10, 20, 50, 100, 125 (valore predefinito), 250, 500, 800 o 1000 Kbit/s.** **Nota:** verificare che ogni apparecchiatura slave dichiarata sulla rete sia configurata singolarmente in modo che il proprio Baudrate sia assolutamente identico alla velocità di rete definita in precedenza. In caso contrario, le comunicazioni sulla rete CANopen non potranno funzionare correttamente.
2	Configurare il periodo di Life-time. Questo parametro definisce il periodo del ciclo delle comunicazioni che verrà implementato nel campo di supervisione di ogni apparecchiatura slave, come descritto al successivo passo 3. **Nota:** non immettere 0 in questo campo.
3	Fare clic una volta nel campo **Supervisione** per configurare le opzioni del protocollo di controllo degli errori di ogni apparecchiatura slave dichiarata nella tabella di rete. **Risultato:** le opzioni di supervisione disponibili supportate dall'apparecchiatura selezionata sono riportate in una casella di riepilogo, come mostrato nell'esempio seguente: 
4	Selezionare il protocollo di controllo degli errori che si desidera utilizzare per la gestione delle comunicazioni tra il modulo master TWDNCO1M e l'apparecchiatura slave selezionata: ● Guard Time ● Heartbeat ● Nessuna
5	Se nella tabella degli slave di rete l'opzione di supervisione è impostata su **Nessuna**, in caso di interruzione della connessione (*) tra questo slave e il modulo master TWDNCO1M, le uscite non torneranno ai valori di sicurezza. (*) questa disconnessione può essere provocata da: ● disconnessione del cavo del bus di espansione che collega il modulo master CANopen TWDNCO1M al controller di base Twido, ● disconnessione dello slave CANopen dal bus CANopen Twido, ● un guasto al cavo del bus, ● un comando Reset di TwidoSoft (In linea → Firmware / Reset), ● un comando di caricamento della configurazione di TwidoSoft (Online → Scarica), ● un comando di trasferimento del firmware verso il modulo master TWDNCO1M tramite TwidoSoft (In linea → Scarica firmware).
6	Premere il pulsante **Applica** per confermare le modifiche e salvare la configurazione di rete nel progetto TwidoSoft.

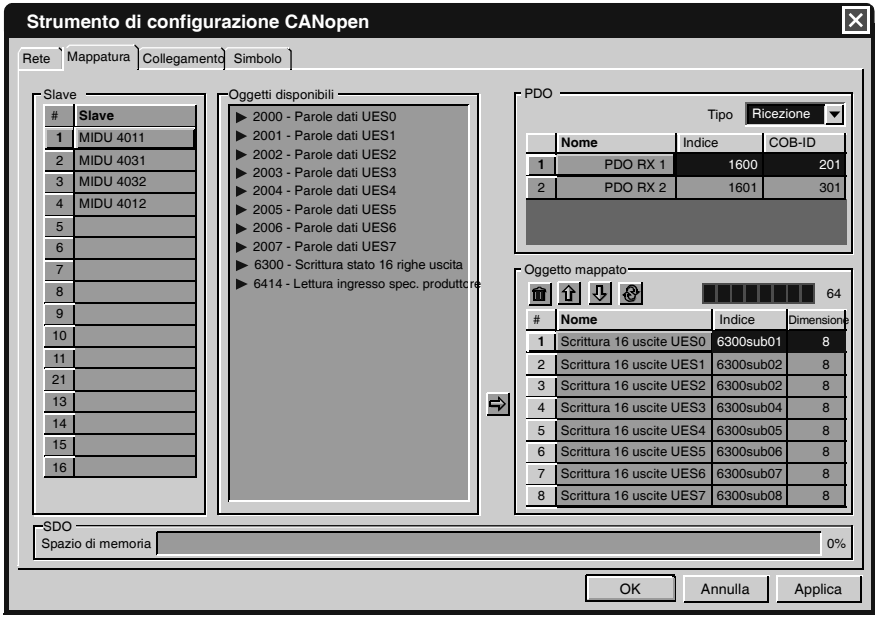
Mappatura oggetti CANopen

Panoramica

La finestra di dialogo **Mappatura** dello strumento di configurazione CANopen permette di configurare i PDO di ogni dispositivo slave dichiarato sulla rete.

Finestra di dialogo Mappatura

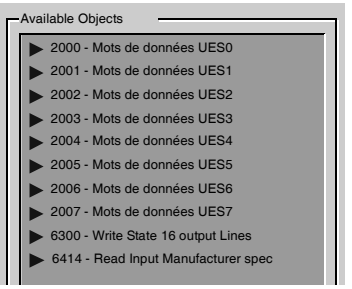
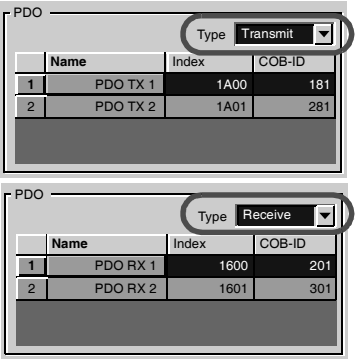
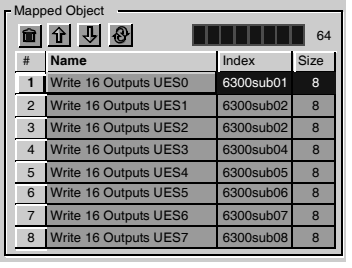
Dal browser dell'applicazione TwidoSoft, fare clic con il pulsante destro del mouse sul nome del modulo master e selezionare **Hardware** → **Bus di espansione** → **TWDNCO1M** → **Configura** e quindi selezionare la scheda **Mappatura** dallo strumento di configurazione CANopen.
Risultato: viene visualizzato lo strumento di configurazione CANopen, come mostrato nella seguente figura.



Mappatura degli oggetti

Per informazioni su come utilizzare la finestra di dialogo Mappatura per configurare i TPDO e gli RPDO di ogni dispositivo slave, procedere come segue:

Passo	Azione
1	Nel riquadro **Slave**, fare clic una volta sul nome del dispositivo e selezionare lo slave di cui si desidera configurare i PDO.
2	**Esempio:** Il modulo DS-401 I/O ha etichetta MIDU 4011. Si noti che i nomi degli slave e gli indirizzi dei nodi sono riportati in questo riquadro esattamente come definiti nella fase precedente della configurazione di rete (vedere *Dichiarazione sulla rete di slave CANopen, p. 261*). 

Passo	Azione
3	**Risultato:** **1.** tutti gli oggetti CANopen supportati dallo slave selezionato sono visualizzati nel riquadro **Oggetti disponibili**, come riportato nell'esempio seguente:  **2.** Il riquadro **PDO** riporta i Transmit-PDOs (PDO TX) predefiniti per lo slave selezionato . L'elenco **Tipo** permette di alternare la visualizzazione mostrando anche i Receive-PDOs (PDO RX) predefiniti. In questo esempio, il modulo di I/O MIDU 4011 DS-401 supporta due Transmit-PDOs (PDO TX) e due Receive-PDOs (PDO RX) come mostrato di seguito:  **3.** La mappatura predefinita di ogni PDO selezionato è riportata anche nel riquadro **Oggetti mappati**. : 

Passo	Azione
4	se desiderato, è possibile personalizzare la mappatura PDO tramite il riquadro **Oggetti mappati**. Un RPDO o un TPDO è un oggetto a 64 byte che può contenere fino a otto oggetti parola a 8 byte o quattro oggetti parola a 16 byte o qualsiasi combinazione dei due tipi di oggetti parola, purché non si superi il limite complessivo di 64 byte del PDO. Per personalizzare la mappatura PDO, proseguire con il passo 5 e successivi e procedere come indicato.
5	Per gli slave desiderati (vedere passo 2), selezionare il PDO di cui si desidera modificare la mappatura dal riquadro **PDO**. **Esempio:** Selezionare il primo Transmit-PDO (PDO TX 1). **Risultato:** nel riquadro Oggetti mappati viene visualizzata la mappatura predefinita dei PDO o la mappatura personalizzata corrente.
6	Per eliminare un oggetto parola inutilizzato dalla struttura di mappatura dei PDO, selezionare l'oggetto parola (indicizzato da 1 a 8) e fare clic sull'icona **Elimina** .
7	Dal riquadro **Oggetti disponibili**, selezionare l'oggetto parola nella famiglia di oggetti da mappare e fare clic sull'icona **Aggiungi**  per aggiungerlo alla struttura degli **Oggetti mappati**. **Nota:** per ripristinare la struttura di mappatura predefinita per il PDO selezionato, fare clic sull'icona **Predefinito** .
8	Per modificare l'indirizzo di un oggetto parola nella struttura dei PDO mappati, utilizzare le icone delle frecce **su e giù** .
9	Premere il pulsante **Applica** per confermare le modifiche apportate alla struttura dei PDO mappati e salvare la mappatura dei PDO nel progetto TwidoSoft.
10	Ripetere i passi da 5 a 9 per ogni mappatura di PDO da configurare.
11	**Note sull'uso della memoria:** ● **Uso della memoria PDO:** Per tenere sotto controllo l'uso della memoria PDO è possibile utilizzare la barra di stato della memoria che si trova nell'angolo in alto a destra del riquadro degli oggetti mappati:  . ● **Uso di memoria SDO aggiuntiva:** I PDO predefiniti e gli oggetti parola non utilizzano memoria SDO aggiuntiva. Sia l'aggiunta sia la rimozione di oggetti parola dalla struttura di mappatura dei PDO, tuttavia, richiede l'uso di memoria di sistema supplementare. L'uso corrente della memoria SDO è visualizzato nella barra di stato che si trova nella parte inferiore della finestra di dialogo Mappatura:  .

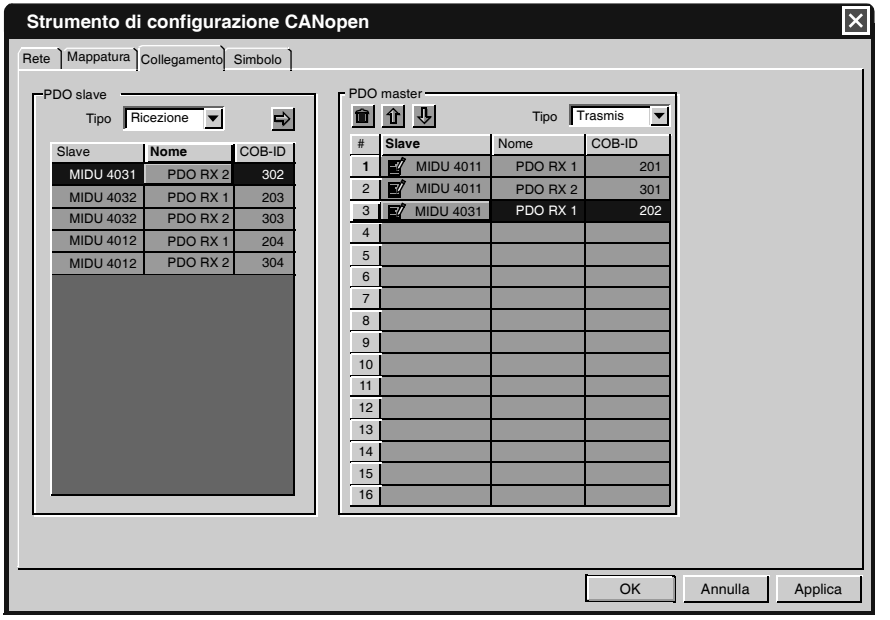
Collegamento oggetti CANopen

Panoramica

La finestra di dialogo **Collegamento** dello strumento di configurazione CANopen consente di definire il collegamento fisico tra i PDO selezionati delle apparecchiature slave e i PDO del modulo master CANopen TWDNCO1M.

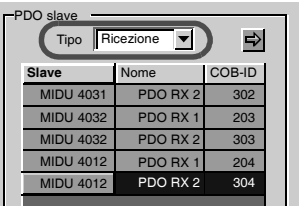
Finestra di dialogo Collegamento

Dal browser dell'applicazione di TwidoSoft, fare clic con il pulsante destro del mouse sul nome del modulo master e selezionare **Hardware** → **Bus di espansione** → **TWDNCO1M** → **Configura**. Selezionare quindi la scheda **Collegamento** nello strumento di configurazione CANopen.
Risultato: viene visualizzato lo strumento di configurazione CANopen, come mostrato nella seguente figura.



Collegamento di oggetti

Per informazioni su come utilizzare la finestra di dialogo Collegamento per definire il collegamento fisico tra i PDO dell'apparecchiatura slave e quelli del modulo master, procedere come segue:

Passo	Azione
1	Nel riquadro **PDO slave**, selezionare il **Tipo:** Receive del PDO oppure Transmit. **Risultato:** tutti gli slave PDO del tipo selezionato vengono visualizzati nel riquadro PDO slave, come mostrato dall'esempio seguente:  **Nota:** se si seleziona Receive oppure Transmit, la visualizzazione passa automaticamente dai PDO slave ai PDO master del tipo opposto: Transmit oppure Receive, rispettivamente.
2	Dal riquadro **PDO slave**, selezionare il PDO da collegare al master CANopen TWDNCO1M e fare clic sull'icona **Aggiungi** per aggiungere il PDO all'elenco dei collegamenti **PDO master**. **Nota:** il master TWDNCO1M supporta al massimo 16 collegamenti TPDO e 16 RPDO.
3	Per modificare l'indice indirizzo di un collegamento PDO nel riquadro PDO master, utilizzare le icone delle frecce di spostamento **su e giù** / .
4	Per eliminare un collegamento PDO non utilizzato nel riquadro PDO master, selezionare il PDO desiderato (indicizzato da 1 a 16) e fare clic sull'icona **Elimina** .
5	Premere il pulsante **Applica** per confermare le modifiche apportate alla struttura dei PDO mappati e salvare la mappatura dei PDO nel progetto TwidoSoft.
6	Ripetere i passi da 1 a 5 per ogni PDO slave da collegare al master CANopen.

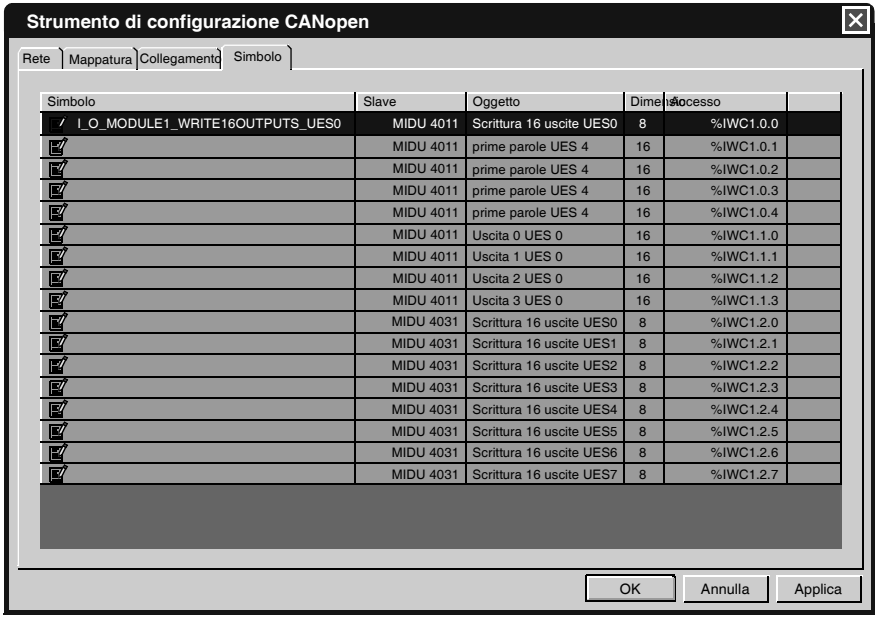
Rappresentazione simbolica degli oggetti CANopen

Panoramica

La finestra di dialogo **Simbolo** consente di definire una rappresentazione simbolica delle variabili associate con il master CANopen.

Finestra di dialogo Simbolo

Dal browser dell'applicazione di TwidoSoft, fare clic con il pulsante destro del mouse sul nome del modulo master e selezionare **Hardware** → **Bus di espansione** → **TWDNCO1M** → **Configura** e scegliere la scheda **Simbolo** dello strumento di configurazione CANopen.
Risultato: viene visualizzato lo strumento di configurazione CANopen, come mostrato nella seguente figura.



Rappresentazione simbolica degli oggetti

Per informazioni su come utilizzare la finestra di dialogo Simbolo per definire i simboli delle variabili oggetto CANopen, procedere come segue:

Passo	Azione
1	Nel campo **Simbolo**, fare doppio clic sull'icona di modifica dei simboli  in corrispondenza della riga per la cui variabile si desidera creare la rappresentazione simbolica. **Risultato:** La casella di testo del simbolo viene attivata e il cursore è allineato sulla destra, al suo interno.
2	Compilare con un nome descrittivo. Un simbolo valido può avere fino a 32 caratteri: sono ammesse solo lettere da A a Z, numeri da 0 a 9 e il carattere di sottolineatura "_" (non sono ammessi i seguenti caratteri "/", "%", spazio e qualsiasi altro carattere speciale). **Nota:** per ulteriori informazioni sulla modifica dei simboli, vedere *Oggetti di simbolizzazione, p. 52*.
3	Premere il pulsante **Applica** per confermare le modifiche alla tabella dei simboli e salvarli nel progetto TwidoSoft.
4	Ripetere i passi da 1 a 3 per ogni variabile che si desidera rappresentare simbolicamente.

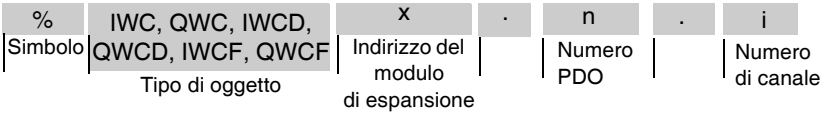
Indirizzamento dei PDO del master CANopen

In breve

Questa sottosezione descrive come eseguire l'indirizzamento degli ingressi e delle uscite PDO del master CANopen.
Per evitare confusione con gli I/O remoti, per la sintassi degli oggetti CANopen si implementa una nuova designazione: ad esempio, %I**WC**.

Illustrazione

Promemoria sui principi di indirizzamento:



Valori specifici

La tabella seguente fornisce i valori specifici per gli oggetti degli slave CANopen.

Parte	Valori	Commento
IWC	-	Immagine degli ingressi PDO fisici.
QWC	-	Immagine delle uscite PDO fisiche.
IWCD	-	Uso uguale a IWC ma in formato parola doppia.
QWCD	-	Uso uguale a QWC ma in formato parola doppia.
IWCF	-	Uso uguale a IWC ma in formato virgola mobile.
QWCF	-	Uso uguale a QWC ma in formato virgola mobile.
x	da 1 a 7	Indirizzo del modulo master CANopen TWDNCO1M sul bus di espansione Twido.
n	da 0 a 15	Numero dei PDO secondo l'indice PDO.
i	da 0 a 7	Numero dei canali secondo il sottoindice PDO.

Esempio

La tabella seguente descrive un esempio di indirizzamento dei PDO.

Oggetto di I/O	Descrizione
%IWC4.1.0	PDO numero 1, ingresso 0 del sottoindice del modulo CANopen situato all'indirizzo 4 sul bus di espansione Twido.

Scambi impliciti

Gli oggetti descritti di seguito sono scambiati in modo implicito, in altre parole, vengono scambiati automaticamente a ogni ciclo del controller.

Programmazione e diagnostica del bus di campo CANopen

Scambi espliciti

Gli oggetti (parole e bit) associati al bus di campo CANopen portano informazioni (ad esempio: funzionamento del bus, stato degli slave e così via) e mettono a disposizione comandi supplementari per eseguire la programmazione avanzata della funzione CANopen.
Questi oggetti vengono scambiati in modo esplicito tra il controller Twido e il modulo master CANopen tramite il bus di espansione:

- su domanda del programma utente tramite l'istruzione: CAN_CMD (vedere di seguito "Panoramica sull'istruzione CAN_CMD"),
- tramite la schermata di debug o la tabella di animazione.

Parole di sistema specifiche riservate master CANopen

Le parole di sistema riservate nel controller Twido per i moduli master CANopen consentono di conoscere lo stato della rete: %SW8x (x=1-7) è riservato per il modulo master CANopen installato all'indirizzo di espansione x sul bus Twido. Vengono utilizzati solo i primi 7 bit di queste parole e in sola lettura.
La tabella seguente presenta i bit utilizzati:

Parole di sistema	Bit	Descrizione
%SW8x (x=1-7)	0	Stato di configurazione del master CANopen (= 1 configurazione corretta, altrimenti 0)
	1	Modalità operativa del master CANopen (= 1 scambio dati abilitato, altrimenti 0)
	2	Sistema in stop (= 1 modalità fuori linea abilitata, altrimenti 0)
	3	Istruzione CAN_CMD completa (= 1 comando completato, altrimenti 0 quando il comando è in esecuzione)
	4	Errore dell'istruzione CAN_CMD (= 1 errore nell'istruzione, altrimenti 0)
	5	Errore di inizializzazione (= 1)
	6	Perdita del messaggio, errore di alimentazione (= 1)

Esempio di uso, per il modulo master CANopen installato all'indirizzo di espansione 1 sul bus Twido):
Prima di utilizzare un'istruzione CAN_CMD, si deve verificare il bit %SW81:X3 per sapere se è in corso un'istruzione: verificare che %SW81:X3 = 1.
Per conoscere l'esito corretto dell'istruzione, verificare che il bit %SW81:X4 sia uguale a 0.

Parole di sistema specifiche riservate slave CANopen

Le parole da %SW20 a %SW27 sono parole di sistema riservate che permettono di conoscere lo stato corrente dei 16 slave CANopen con indirizzi di nodo compresi tra 1 e 16. Il contenuto di queste parole di memoria è in sola lettura.
La seguente tabella descrive le parole di sistema da %SW20 a %SW27:

Parole di sistema	Indirizzo nodo (numero di slave)		Contenuto/descrizione della parola
	Bit [0-7]	Bit [8-15]	
%SW20	1	2	Quando %SW2x assume il seguente valore: ● = 1 => modulo non previsto presente sulla rete. Si è segnalato come "non esente da errori" prima di essere rimosso dalla rete. ● = 2 => Stato di funzionamento del nodo (il modulo è nello stato di funzionamento): - "senza errori". ● = 3 => Stato di funzionamento del nodo (il modulo è nello stato di funzionamento): - "non esente da errori". ● = 4 => Stato preoperativo del nodo (il modulo è in modalità preoperativa): - solo moduli previsti (quelli dichiarati e previsti come da tabella di configurazione); - il modulo può essere posto in stato di funzionamento; - "senza errori". ● = 5 => Stato preoperativo del nodo (il modulo è nello stato preoperativo): - solo moduli previsti (quelli dichiarati e previsti come da tabella di configurazione); - il modulo può essere posto in stato operativo; - "con errore".
%SW21	3	4	
%SW22	5	6	
%SW23	7	8	
%SW24	9	10	
%SW25	11	12	
%SW26	13	14	
%SW27	15	16	

Parole di sistema	Indirizzo nodo (numero di slave)		Contenuto/descrizione della parola
	Bit [0-7]	Bit [8-15]	
			• = 6 => Stato preoperativo del nodo (il modulo è nello stato preoperativo): - solo moduli previsti (quelli dichiarati e previsti come da tabella di configurazione); - il modulo è presente ma il suo stato corrente non consente di passarlo nello stato operativo; - "senza errori". • = 7 => Stato preoperativo del nodo (il modulo è nello stato preoperativo): - solo moduli previsti (quelli dichiarati e previsti come da tabella di configurazione); - il modulo è presente ma il suo stato corrente non consente di passarlo nello stato operativo; - "con errore". • = 8 => Modulo errato (è stato identificato un modulo con ID dispositivo diverso da quello previsto): - "senza errori". • = 9 => Modulo errato (è stato identificato un modulo con ID dispositivo diverso da quello previsto): - "con errore". • = 10 => Errore di configurazione dello slave (il modulo ha risposto alla richiesta di scrittura SDO della tabella di comando SDO, con la conferma di un errore, o non ha rispettato le regole del protocollo SDO): - "senza errori". • = 11 => Errore di configurazione dello slave: - "con errore". • = 12 => Modulo mancante/ Timeout di controllo errore / Timeout SDO (un modulo che era stato configurato non è più disponibile, non risponde durante il funzionamento o non risponde all'accesso dell'SDO): - "senza errori".

Parole di sistema	Indirizzo nodo (numero di slave)		Contenuto/descrizione della parola
	Bit [0-7]	Bit [8-15]	
			• = 13 => Modulo mancante/ Timeout di controllo errore / Timeout SDO (un modulo che era stato configurato non è più disponibile, non risponde durante il funzionamento o non risponde all'accesso dell'SDO): - "con errore". • = 14 => Modulo non previsto (è stato rilevato un modulo che non è presente nella tabella di configurazione): - "senza errori". • = 15 => Modulo non previsto (è stato rilevato un modulo che non è presente nella tabella di configurazione): - "con errore".

Panoramica sull'istruzione CAN_CMD

Per ogni programma utente, l'istruzione CAN_CMD permette di programmare la rete e di ottenere la diagnostica degli slave. I parametri dell'istruzione vengono impostati tramite parole interne (parole di memoria) **%MW**x.
La sintassi dell'istruzione è la seguente:
CAN_CMDn **%MW**x**:**l
Legenda:

Simbolo	Descrizione
n	Indirizzo di espansione del modulo master CANopen sul bus Twido (da 1 a 7).
x	Numero della prima parola interna (parola di memoria) specificato nel parametro (da 0 a 254).
l	Lunghezza dell'istruzione in numero di parole (2).

Uso dell'istruzione CAN_CMD

L'istruzione CAN_CMD consente di programmare e gestire la rete CANopen e di eseguire controlli di diagnostica sui singoli dispositivi slave. I parametri dei comandi sono trasmessi tramite parole di memoria %MWx.
La tabella che segue descrive l'azione dell'istruzione CAN_CMD in base al valore dei parametri da %MW(x) a %MW(x+5) a seconda dei casi:

<table>
<tr><th rowspan="2">%MWx</th><th colspan="2">%MWx+1</th><th colspan="2">%MWx+2</th><th colspan="2">%MWx+3</th><th colspan="2">%MWx+4</th><th colspan="2">%MWx+5</th><th rowspan="2">Azione</th></tr>
<tr><th>Bit [0-7]</th><th>Bit [8-15]</th><th>Bit [0-7]</th><th>Bit [8-15]</th><th>Bit [0-7]</th><th>Bit [8-15]</th><th>Bit [0-7]</th><th>Bit [8-15]</th><th>Bit [0-7]</th><th>Bit [8-15]</th></tr>
<tr><td>1</td><td colspan="2">0</td><td colspan="8" rowspan="4">—</td><td>Ripristina comunicazione CANopen.</td></tr>
<tr><td>1</td><td colspan="2">1</td><td>Ripristina nodi CANopen</td></tr>
<tr><td>2</td><td colspan="2">0</td><td>Passa dalla modalità operativa alla modalità preoperativa.</td></tr>
<tr><td>2</td><td colspan="2">1</td><td>Passa alla modalità operativa.</td></tr>
<tr><td rowspan="6">3 o 4</td><td colspan="2"></td><td colspan="8" rowspan="2"></td><td>3 => Comando Start Read SDO.
4 => Comando Start Write SDO.</td></tr>
<tr><td colspan="2">Nodo</td><td>Nodo = 1-16 => Indirizzo nodo</td></tr>
<tr><td colspan="2"></td><td colspan="2">Indice</td><td colspan="6"></td><td>Indice oggetto PDO.</td></tr>
<tr><td colspan="4"></td><td>Len</td><td>Sub</td><td colspan="4"></td><td>Sub = 0-255 => Sottoindice oggetto
Len = Lunghezza dei dati nel byte</td></tr>
<tr><td colspan="6"></td><td colspan="2">Dato 1</td><td></td><td>Payload basato sul valore del campo lunghezza (Len)</td></tr>
<tr><td colspan="8"></td><td>Dato 2</td><td>Payload basato sul valore del campo lunghezza (Len)</td></tr>
</table>

Nota: lo stato del bus è aggiornato a ogni scansione del controller. Il risultato dell'istruzione CAN_CMD di lettura del bus, tuttavia, è disponibile solo al termine del ciclo successivo del controller.

Esempi di programmazione dell'istruzione CAN_CMD

Esempio 1:
Per forzare il master CANopen che si trova all'indirizzo 1 sul bus di espansione Twido e passare alla modalità di inizializzazione:

```
LD 1
[%MW0 := 16#0001 ]
[%MW1 := 16#0001 ]
LD %SW81:X3          // Se non vi è alcuna istruzione CAN_CMD in esecuzione e proseguire
[CAN_CMD1 %MW0:2]    // Per forzare il master CANopen a passare alla modalità di inizializzazione
LD %SW81:X4          // (opzionale) Per sapere se l'istruzione CAN_CMD è stata completata correttamente prima di inviarne una nuova.
```

Esempio 2:
Per leggere la seguente variabile: SDO_Slave:1_index:24576_sub-index:1_length:4

```
LD 1
[%MW6 := %MW4]       // Memorizza il risultato dell'ultimo comando SDO
[%MW7 := %MW5]       // Memorizza il risultato dell'ultimo comando SDO

LD %SW81:X3  // Se non vi sono istruzioni CAN_CMD in esecuzione e continuare
[%MW0 := 16#0003]
[%MW1 := 16#0001]    // Lettura SDO al nodo di indirizzo 1
[%MW2 := 16#6000]    // Accesso al numero indice 24576
[%MW3 := 16#0104]    // Accesso al numero di sottoindice 1 e valore lunghezza 4
[CAN_CMD1 %MW0:6] // Avvio comando SDO
```

Esempio 3:
Per scrivere la seguente variabile: SDO_Slave:1_index:24576_sub-index:1_length:4

```
LD 1
[%MW0 := 16#0004]
[%MW1 := 16#0001]    // Scrittura SDO al nodo di indirizzo 1
[%MW2 := 16#6000]    // Accesso al numero indice 24576
[%MW3 := 16#0104]    // Accesso al numero di sottoindice 1 e valore lunghezza 4
[%MW4 := 16#1234]    // Valori dato 1
[%MW5 := 16#1234]    // Valori dato 2
LD %SW81:X3  // Se non vi sono istruzioni CAN_CMD in esecuzione e continuare
[CAN_CMD1 %MW0:6] // Avvio comando SDO
```

Configurazione del gateway Ethernet TwidoPort

11

In breve

Contenuto del capitolo

Questo capitolo fornisce informazioni sulla configurazione software del modulo gateway Ethernet ConneXium TwidoPort.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
11.1	Configurazione normale e connessione di TwidoPort	283
11.2	Configurazione Telnet di TwidoPort	290
11.3	Funzioni di comunicazione	304

11.1 Configurazione normale e connessione di TwidoPort

In breve

Contenuto della sezione

Questa sezione fornisce informazioni su come eseguire una normale configurazione del modulo ConneXium TwidoPort con il programma applicativo TwidoSoft, sulla connettività del modulo e sulla configurazione BootP.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Configurazione normale con TwidoSoft	284
Configurazione BootP	289

Configurazione normale con TwidoSoft

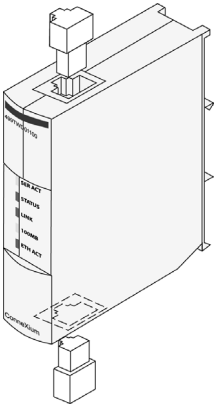
Informazioni preliminari

Se si dispone di TwidoSoft v. 3.0 o successiva, configurare TwidoPort procedendo come segue:

Nota: funzione Plug 'n play
Quando si configura TwidoPort con TwidoSoft, l'impostazione IP di TwidoPort viene memorizzata nel controller Twido. I tecnici della manutenzione, pertanto, possono scambiare i moduli TwidoPort senza impostare altri parametri.
Per utilizzare la funzionalità plug 'n play, occorre avere a disposizione TwidoSoft versione 3.0 o successiva e aggiornare il firmware Twido alla versione 3.0 o successiva. Per configurare in modo manuale TwidoPort con versioni precedenti di TwidoSoft, utilizzare Telnet.

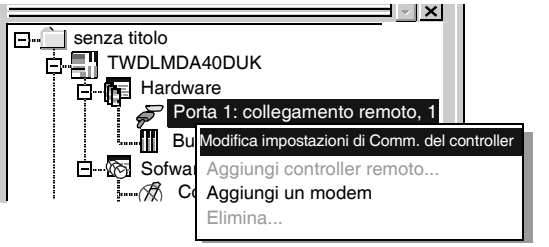
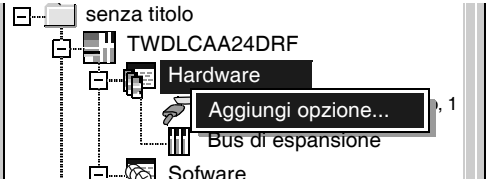
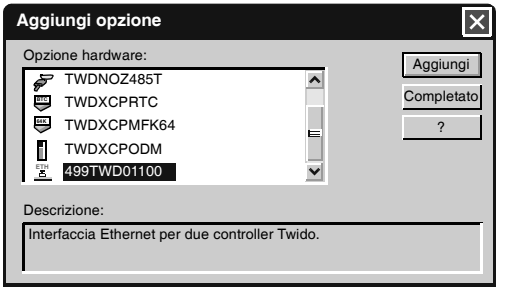
Installazione del modulo 499TWD01100 TwidoPort

Per installare TwidoPort su un sistema composto da controller Twido (guida DIN o montaggio a pannello) e collegarlo al bus interno del controller Twido, procedere come segue:

Passo	Descrizione	Azione
1	Preparazione dell'installazione	Consultare il *manuale di riferimento hardware dei controller programmabili Twido (TWD USE 10AE)* per istruzioni su: ● posizioni di montaggio corrette per i moduli Twido, ● aggiunta e rimozione dei componenti Twido da una guida DIN, ● montaggio diretto sulla superficie di un pannello, ● spazio minimo intorno ai moduli in un pannello di controllo.
2	Montaggio del modulo 499TWD01100 TwidoPort.	Installare il modulo su una guida DIN o su un pannello. Per ulteriori informazioni, vedere (Vedere TwdoHW Guida di riferimento dell'hardware).
3	Messa a terra	Collegare un filo di messa a terra alla morsettiera a vite M3 sul fondo di TwidoPort.
4	Connessioni seriali ed Ethernet **Spina superiore: da Twido (seriale)**  **Spina inferiore: da Ethernet, cavo diretto o incrociato**	Collegare la spina modulare del cavo di collegamento da TwidoPort a Twido fornito alla porta seriale di TwidoPort e collegare l'altra estremità alla porta seriale RS-485 del controller Twido. Collegare la spina RJ-45 di un cavo di rete Ethernet standard non fornito alla porta Ethernet di TwidoPort.

Dichiarazione del modulo 499TWD01100 TwidoPort

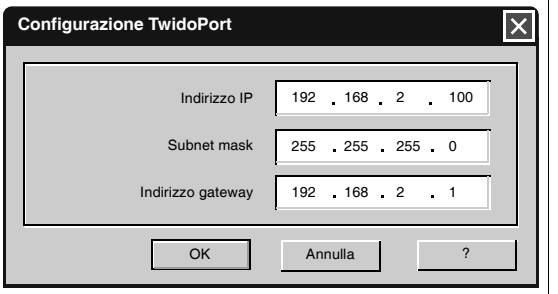
La tabella seguente mostra i vari passaggi della dichiarazione del modulo 499TWD01100 TwidoPort.

Passo	Azione	Commento
1	Quando si utilizza TwidoSoft versione 3.0 o successiva, è possibile configurare le opzioni di comunicazione del controller Twido facendo clic con il pulsante destro del mouse su **Porta 1: xxxxxx , 1** → **Modifica impostazioni di Com. del Controller...**, utilizzando il browser dell'applicazione TwidoSoft. (Vedere nota 1).	
2	Dalla finestra di dialogo **Impostazione delle comunicazioni del Controller**, impostare il protocollo di comunicazione su **Modbus.**	Si tenga presente che se il protocollo di comunicazione non è impostato su Modbus, non è possibile aggiungere il modulo TwidoPort 499TWD01100 all'hardware Twido.
3	Configurare i parametri di comunicazione Modbus.	Perché sia supportata la funzione di velocità automatica di TwidoPort, è necessario che la porta Modbus RS-485 del controller Twido sia configurata su **9600, 19200 o 38400 baud**. (Vedere le note 1 e 2).
4	Nel browser dell'applicazione TwidoSoft, fare clic con il pulsante destro del mouse su **Hardware** → **Aggiungi opzione...**.	
5	Quando viene visualizzata la finestra di dialogo **Aggiungi opzione**: • Selezionare **499TWD01100** • Fare clic su **Aggiungi.** • A questo punto, è possibile continuare ad aggiungere altri moduli opzionali supportati dal controller Twido. **Nota:** è possibile configurare solo un modulo TwidoPort 499TWD01100. • Fare clic su **Completato.**	Tutti i controller Twido sono supportati, ad eccezione di TWDLCAE40DRF con interfaccia Ethernet incorporata. 
Nota 1	**Su Twido è possibile utilizzare qualsiasi porta Modbus RS-485.**	
Nota 2	**Per ottenere la velocità automatica iniziale più elevata, selezionare 19200-8-N-1 con indirizzo Modbus Twido 1.**	

Configurazione del modulo TwidoPort 499TWD01100

Nota: è possibile configurare i parametri Ethernet di TwidoPort solo quando il programma applicativo TwidoSoft si trova in modalità non in linea.

Per configurare i parametri Ethernet di TwidoPort, procedere come segue:

Passo	Azione	Commento
Introduzione	Per ulteriori informazioni sui parametri IP (indirizzo IP, subnet mask e indirizzo gateway), vedere *Indirizzamento IP, p. 165* e *Indirizzi IP privati, p. 169*.	
1	Fare clic con il pulsante destro del mouse sull'icona di TwidoPort nel browser dell'applicazione per configurare i parametri IP di TwidoPort	**Risultato:** viene visualizzata la finestra di dialogo **Configurazione Ethernet**, come riportato nell'esempio seguente. 
2	Immettere l'**Indirizzo IP** statico di TwidoPort in formato decimale separato da punti. (Vedere le *note 1 e 2*).	**Attenzione:** per garantire una buona comunicazione tra i dispositivi, è necessario che gli indirizzi IP del PC che esegue l'applicazione TwidoSoft e del modulo TwidoPort condividano lo stesso ID di rete.
Nota 1	**Per ottenere i parametri IP validi per la rete, rivolgersi all'amministratore di rete o di sistema.**	
Nota 2	**Per garantire una buona comunicazione di rete, ogni dispositivo collegato deve avere un indirizzo IP univoco. Quando è collegato alla rete, il modulo TwidoPort verifica che non esistano indirizzi IP doppi. Se nella rete viene rilevato un indirizzo IP doppio, il LED STATUS lampeggia per 4 volte di seguito. In questo campo occorre quindi immettere un nuovo indirizzo IP non doppio.**	
Nota 3	**Salvo requisiti speciali del modulo TwidoPort, usare il valore di subnet mask predefinito.**	
Nota 4	**Se la rete non comprende alcun dispositivo gateway, immettere nel campo Indirizzo Gateway l'indirizzo IP del modulo TwidoPort.**	

Passo	Azione	Commento
3	Immettere il valore di **Subnetwork mask** valido assegnato a TwidoPort dall'amministratore di rete. Si noti che questo campo non può rimanere vuoto; è quindi necessario immettere un valore. (Vedere le *note 1 e 3*).	**Attenzione:** per garantire una buona comunicazione tra i dispositivi, il valore di subnet mask configurato sul PC che esegue l'applicazione TwidoSoft e quello di TwidoPort devono corrispondere. Per impostazione predefinita, l'applicazione TwidoSoft calcola automaticamente e visualizza un valore di subnet mask predefinito basato sull'IP di classe fornito nel campo precedente Indirizzo IP. I valori predefiniti di subnet mask, a seconda della categoria dell'indirizzo IP di rete di TwidoPort, seguono questa regola: Rete classe A -> Valore di subnet mask predefinito: 255.0.0.0 Rete classe B -> Valore di subnet mask predefinito: 255.255.0.0 Rete classe C -> Valore di subnet mask predefinito: 255.255.255.0
4	Immettere l'indirizzo IP del **Gateway**. (Vedere le *note 1 e 4*).	Nella LAN, il gateway deve trovarsi sullo stesso segmento del modulo TwidoPort. Questa informazione viene generalmente fornita dall'amministratore di rete. Si noti che l'applicazione non fornisce alcun valore predefinito; occorre pertanto immettere un indirizzo di gateway valido in questo campo.
5	Confermare la configurazione e scaricarla nel controller Twido.	
Nota 1	**Per ottenere i parametri IP validi per la rete, rivolgersi all'amministratore di rete o di sistema.**	
Nota 2	**Per garantire una buona comunicazione di rete, ogni dispositivo collegato deve avere un indirizzo IP univoco. Quando è collegato alla rete, il modulo TwidoPort verifica che non esistano indirizzi IP doppi. Se nella rete viene rilevato un indirizzo IP doppio, il LED STATUS lampeggia per 4 volte di seguito. In questo campo occorre quindi immettere un nuovo indirizzo IP non doppio.**	
Nota 3	**Salvo requisiti speciali del modulo TwidoPort, usare il valore di subnet mask predefinito.**	
Nota 4	**Se la rete non comprende alcun dispositivo gateway, immettere nel campo Indirizzo Gateway l'indirizzo IP del modulo TwidoPort.**	

Configurazione BootP

Processo BootP

TwidoPort attende una risposta dal server BootP entro due minuti dalla trasmissione della richiesta BootP. Se non la riceve, TwidoPort assume la configurazione IP predefinita creata in base all'indirizzo MAC della struttura:

85	16	MAC[4]	MAC[5]

Indirizzo MAC

L'indirizzo MAC ha la seguente struttura:
`MAC[0] MAC[1] MAC[2] MAC[3] MAC[4] MAC[5].`

Se, ad esempio, l'indirizzo MAC è `0080F4012C71`, l'indirizzo IP predefinito sarà `85.16.44.113`.

11.2 Configurazione Telnet di TwidoPort

In breve

Contenuto della sezione

Questa sezione descrive come configurare il modulo ConneXium TwidoPort con una sessione Telnet.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Introduzione alla configurazione Telnet	291
Menu principale di Telnet	292
Impostazioni IP/Ethernet	293
Configurazione dei parametri seriali	294
Configurazione del gateway	295
Configurazione della sicurezza	296
Statistiche Ethernet	297
Statistiche seriali	298
Salvataggio della configurazione	299
Ripristino delle impostazioni predefinite	300
Aggiornamento del firmware TwidoPort	301
Se si dimentica la password e/o la configurazione IP	303

Introduzione alla configurazione Telnet

Panoramica sulla configurazione Telnet

È necessario configurare TwidoPort con una sessione Telnet (mediante un client Telnet compatibile VT100) nei casi in cui non è disponibile una configurazione Twido specifica oppure quando la richiesta BootP non riceve risposta dopo due minuti (nel qual caso viene implementato l'indirizzo IP predefinito).

Preparazione per la configurazione Telnet

Nota: Requisiti Telnet per TwidoPort
Quando si esegue la configurazione di TwidoPort con Telnet, verificare quanto segue:

- TwidoPort è fornito con alimentazione (da un controller Twido) tramite la connessione seriale.
- L'**`eco locale`** di Telnet è **`disattivato`**.

Per utilizzare Telnet, aggiungere l'indirizzo IP predefinito o l'indirizzo IP configurato di TwidoPort alla tabella di instradamento del PC tramite il comando:

```
C:\> route add 85.0.0.0 mask 255.0.0.0 local_IP_address_of_PC
```

Esempio:
Se l'indirizzo IP del PC è `192.168.10.30` e l'indirizzo IP predefinito o configurato di TwidoPort è `85.16.44.113`, il comando completo sarà:

```
C:\> route add 85.0.0.0 mask 255.0.0.0 192.168.10.30
```

Menu principale di Telnet

Avvio del menu principale di Telnet

Quando si avvia una sessione Telnet (ad esempio digitando `telnet 85.16.44.113` al prompt di comando oppure utilizzando Windows™ Hyperterminal™), il menu principale di Telnet viene visualizzato dopo aver premuto **`Invio`**:

```
        Telemecanique 499 TWD 01 100 Configuration and Diagnostics
                (c) 2004 Schneider Automation Inc

1) IP/Ethernet Settings
                    IP Source: DEFAULT
                IP Address: 85.16.44.113
            Default Gateway: 85.16.44.113
                    Netmask: 0.0.0.0
        Ethernet Frame Type: ETHERNETII

2) Serial Configuration
        Baud Rate: 19200
        Data Bits: 8
           Parity:  NONE
        Stop Bits: 1
         Protocol:  RTU

3) Gateway Configuration
          Slave Address Source: UNIT_ID
              Gateway Mode: SLAVE
                MB Broadcasts: ENABLED

4) Security Configuration

5) Ethernet Statistics

6) Serial Statistics

Commands: D)efault settings, S)ave, F)irmware Upgrade, Q)uit without save
     Select Command or Parameter(1..6) to change:
```

Impostazioni IP/Ethernet

Configurazione delle impostazioni IP/ Ethernet

Per modificare le impostazioni IP/Ethernet, procedere come segue:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare (digitare) **1** per modificare l'origine IP in **MEMORIZZATA** e premere ***Invio.***	È possibile che l'origine IP sia già **MEMORIZZATA**.
3	Impostare manualmente i parametri IP desiderati. Vedere *Impostazioni Ethernet di TwidoPort* dopo questa tabella.	Gli altri parametri includono: • Indirizzo IP • Gateway predefinito • Netmask • Tipo frame Ethernet
4	Selezionare **R** e premere ***Invio.***	Viene visualizzato il menu principale di Telnet. Per aggiornare la schermata, premere ***Invio***.

IP Source

L'opzione **Origine IP** indica la posizione da cui proviene la configurazione IP:

- MEMORIZZATA dalla memoria flash locale.
- SERVER dal server BootP.
- TWIDO dal controller Twido.

L'indirizzo IP predefinito (DEFAULT) proviene dall'indirizzo MAC. Per definizione, l'opzione predefinita non è selezionabile.

Nota: una configurazione IP valida del controller Twido sostituisce la selezione dell'utente.

Esempio di impostazioni Ethernet

La figura seguente mostra un esempio di impostazioni Ethernet di TwidoPort:

```
Telemecanique 499 TWD 01 100 Configuration and Diagnost
        (c) 2004 Schneider Automation Inc

IP/Ethernet Settings
-----------------------------------------------------
        1)IP Source: DEFAULT
        2)IP Address: 85.16.44.113
        3)Default Gateway: 85.16.44.113
        4)Netmask: 0.0.0.0
        5)Ethernet Frame Type: ETHERNET2
-----------------------------------------------------
Commands: R)eturn to Main Menu
Select Command or Parameter(1..N) to change:
```

Configurazione dei parametri seriali

Informazioni preliminari

Nota: in circostanze normali, non è necessario configurare i parametri seriali di TwidoPort perché il modulo supporta un algoritmo di velocità automatica che elimina il bisogno di ricorrere alla configurazione seriale.

Configurazione dei parametri seriali

Per configurare i parametri seriali di TwidoPort:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare (digitare) `2` per modificare le impostazioni seriali.	Vedere la figura seguente.
3	Verificare o azzerare le impostazioni.	Gli altri parametri includono: • `Velocità` • `Bit di dati` • `Parità` • `Bit di stop` • `Protocollo`
4	Selezionare `R` e premere ***`Invio`***.	Viene visualizzato il menu principale di Telnet. Per aggiornare la schermata, premere ***`Invio`***.

Esempio di impostazioni seriali

La figura seguente mostra un esempio di impostazioni seriali di TwidoPort:

```
Telemecanique 499 TWD 01 100 Configuration and Diagnostics
        (c) 2004 Schneider Automation Inc

 Serial Configuration
        1) Baud Rate:  19200
        2) Data Bits:  8
        3)    Parity:  NONE
        4) Stop Bits:  1
            Protocol:  RTU

Commands: R)eturn to Main Menu
Select Command or Parameter(1..N) to change:
```

Configurazione del gateway

Informazioni preliminari

Nota: In generale, non è necessario configurare i parametri del gateway TwidoPort.

Configurazione dei parametri del gateway

Per configurare il gateway TwidoPort:

Passo	Azione	Commento	
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).	
2	Selezionare (digitare) 3 per modificare i parametri del gateway.	Vedere la figura seguente.	
3	Sono disponibili i seguenti parametri gateway:		
	(1) Origine indirizzo slave	FIXED	Se l'origine dell'indirizzo slave è **FIXED**, impostare l'indirizzo sul valore dell'indirizzo Modbus del controller Twido. Gli indirizzi validi sono compresi nell'intervallo da 1 a 247.
		UNIT_ID	Viene utilizzato l'ID unità del pacchetto dati Modbus/TCP.
	(2) Modalità gateway	SLAVE	È l'unica opzione disponibile per questa versione.
	(3) Broadcast MB	DISABLED	Sulla porta seriale TwidoPort non vengono inviati messaggi broadcast.
		ENABLED	I messaggi broadcast sono inviati dalla porta seriale del controller Twido. *(Vedere la nota più avanti)*
4	Selezionare **R** e premere ***Invio.***	Viene visualizzato il menu principale di Telnet. Per aggiornare la schermata, premere ***Invio***.	
Nota	**Twido non supporta i messaggi Modbus broadcast.**		

Esempio di impostazioni del gateway

La figura seguente mostra un esempio di impostazioni del gateway TwidoPort:

```
Telemecanique 499 TWD 01 100 Configuration and Diagnostics
        (c) 2004 Schneider Automation Inc

Gateway Configuration
        1) Slave Address Source: UNIT_ID
        2) Slave Address: 20
        3) Gateway Mode: SLAVE
        4) MB Broadcasts: ENABLED

Commands: R)eturn to Main Menu
Select Command or Parameter(1..4) to change: _
```

Configurazione della sicurezza

Configurazione delle impostazioni di sicurezza

Per modificare la password predefinita, procedere come segue:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare (digitare) **4** e premere ***Invio.***	Viene visualizzata la schermata Configurazione della sicurezza.
3	Selezionare **C** e premere ***Invio.***	
4	Immettere la password precedente.	La password precedente è nota agli utenti autorizzati. La password predefinita è **USERUSER**.
5	Immettere la nuova password.	Digitare nuovamente la nuova password. *(Vedere la nota più avanti)*
6	Immettere nuovamente la nuova password.	Per le password utilizzabili, vedere la nota più avanti.
7	Selezionare **R** e premere ***Invio.***	Viene visualizzato il menu principale di Telnet. Per aggiornare la schermata, premere ***Invio***.
Nota	**Informazioni sulle password:** ● **lunghezza minima: 4 caratteri** ● **lunghezza massima: 10 caratteri** ● **caratteri ammessi: da 0 a 9, da a a z, da A a Z (senza spazi).**	

Statistiche Ethernet

Visualizzazione delle statistiche Ethernet

Per visualizzare le statistiche Ethernet di TwidoPort:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare (digitare) `5` per visualizzare la schermata `Statistiche del modulo Ethernet`.	Vedere la figura che segue questa tabella.
3	Premere `Invio` per aggiornare la schermata.	
4	Premere `C` per cancellare le statistiche e premere ***`Invio`***.	Tutti i contatori vengono azzerati.
5	Selezionare `R` e premere ***`Invio.`***	Viene visualizzato il menu principale di Telnet. (Per aggiornare la schermata, premere ***`Invio`***).

Schermata Ethernet Module Statistics.

La schermata `Ethernet Module Statistics` di TwidoPort:

```
          Telemecanique 499 TWD 01 100 Configuration and Diagnostics
                    (c) 2004 Schneider Automation Inc
                    ETHERNET MODULE STATISTICS
-------------------------------------------------------------------------------
Status: 0x9103                          IP Address: 192.168.1.141
System Log Entry: No                    Mac Address: 0:80:f4:0:4c:18
Transmit Speed: 100BASE-T               Subnet Mask: 255.255.0.0
Full/Half Duplex: Half Duplex           Gateway Address: 192.168.1.1
-------------------------------------------------------------------------------
Transmit Statistics     Receive Statistics      Functioning Errors

Transmits: 63           Receives: 532             Missed Packets: 0
Transmit Retries: 0     Framing Errors: 0         Collision Errors: 0
Lost Carrier: 0         Overflow Errors: 0        Transmit Timeouts: 0
Late Collision: 0       CRC Errors: 0             Memory Errors: 0
Tx Buffer Errors: 0     Rx Buffer Errors: 0       Net Interface Restarts: 0
SILO Underflow: 0
-------------------------------------------------------------------------------
Broadcast Packets Received: 37          Multicast Packets Received: 7
-------------------------------------------------------------------------------
Commands: [Enter] to Refresh, C)lear Statistics, R)eturn to Main Menu
```

Statistiche seriali

Visualizzazione delle statistiche seriali

Per visualizzare le statistiche seriali di TwidoPort:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare (digitare) `6` per visualizzare la schermata **`Statistiche seriali`** e premere **`Invio`**.	Vedere la figura che segue questa tabella. Le statistiche seriali vengono aggiornate.
3	Premere `C` per cancellare le statistiche e premere ***`Invio`***.	Tutti i contatori vengono azzerati.
4	Selezionare `R` e premere ***`Invio.`***	Viene visualizzato il menu principale di Telnet. Per aggiornare la schermata, premere ***`Invio`***.

Schermata Serial Statistics

TwidoPort **`schermata Statistiche seriali`**:

```
                Telemecanique 499 TWD 01 100 Configuration and Diagnostics
                        (c) 2004 Schneider Automation Inc

----------------------- SERIAL STATISTICS --------------------------------

Serial Bus Statistics
        Bus Message Count: 8284
        Bus Comm. Error Count: 0
Modbus Slave Statistics
        Slave Message Count: 4142
        Slave Exception Error Count: 3187
        Slave No Response Count: 0

------------------------------------------------------------------------

Commands: [Enter] to Refresh, C>lear Statistics, R>eturn to Main Menu
```

Salvataggio della configurazione

Salvataggio della configurazione

Per salvare le modifiche della configurazione, digitare la relativa password:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare `S` e premere ***`Invio.`***	
3	Immettere la password di configurazione.	La password predefinita è **`USERUSER`**). *(Vedere la nota più avanti)*
Nota	**Per ulteriori informazioni su come impostare una password di sicurezza personalizzata, vedere *Configurazione della sicurezza, p. 296*.**	

Schermata di conferma Save Configuration

TwidoPort - **`Schermata di conferma Salva configurazione`**:

```
                Telemecanique 499 TWD 01 100 Configuration and Diagnostics
                        (c) 2004 Schneider Automation Inc
                        SAVE CONFIGURATION
--------------------------------------------------------------------------------

         Configuration successfully stored to Twido.
         Reboot your module for the new Configuration to be in effect.

         Rebooting in 5 Seconds. You will lose your telnet connection.

Connection to host lost.
```

Ripristino delle impostazioni predefinite

Ripristino delle impostazioni predefinite

Per ripristinare le impostazioni predefinite di TwidoPort:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare `D` per visualizzare la schermata `Configurazione predefinita`.	Vedere la figura che segue questa tabella.
3	Premere `Invio.`	Premere `Invio.` per visualizzare il menu principale.
4	Salvare la configurazione predefinita.	Vedere Salvataggio della configurazione (Vedi *Salvataggio della configurazione, p. 299*).

Schermata Default Configuration

TwidoPort - `Schermata Configurazione predefinita`:

```
          Telemecanique 499 TWD 01 100 Configuration and Diagnostics
                  (c) 2004 Schneider Automation Inc
                  DEFAULT CONFIGURATION
--------------------------------------------------------------------------------

  IP Address: 192.168.2.102
  Gateway Address: 192.168.2.102
  Subnet Mask: 255.255.0.0
  Frame Type: Ethernet II

  Serial Mode: 19200-8-N-1

  Gateway Mode: Modbus/RTU Slave Attached
                Broadcasts Disabled, Slave Address Source=Unit ID

  Configuration Password: USERUSER

  You must (S)ave the configuration to make it active.

  Returning to Main Menu in 2 Seconds, Hit Enter to refresh._
```

Aggiornamento del firmware TwidoPort

Informazioni preliminari

Nota:
1. Prima di iniziare l'aggiornamento del firmware secondo queste istruzioni, è necessario procurarsi una versione più recente del firmware TwidoPort.
2. Arrestare il processo prima di eseguire l'aggiornamento del firmware.
3. La comunicazione Modbus non è disponibile durante la procedura di aggiornamento del firmware.

Aggiornamento del firmware

Per aggiornare il firmware corrente di TwidoPort con l'ultima versione, procedere come segue:

Passo	Azione	Commento
1	Avviare una sessione Telnet.	Utilizzare le suddette istruzioni per aprire il menu principale Telnet (Vedi *Menu principale di Telnet , p. 292*).
2	Selezionare (digitare) **F** per avviare l'aggiornamento del firmware.	Cinque secondi dopo aver avviato l'aggiornamento del firmware selezionando **F**, TwidoPort si azzera e la connessione Telnet è interrotta.
3	Nella riga di comando, digitare: **ftp** seguito dall'indirizzo IP di TwidoPort.	Ad esempio: **ftp 85.16.44.113**
4	Immettere: **ftptwd**	Alla richiesta del nome di collegamento.
5	Immettere: **cd fw**	In tal modo, si accede alla directory **fw**.
6	Immettere: **put App.out** . *(Vedere le note 1 e 2).*	Un messaggio conferma che il collegamento ftp si è svolto correttamente. *(Vedere nota 3).*
Nota 1	**I nomi dei file fanno differenza tra maiuscole e minuscole.**	
Nota 2	**Verificare che App.out si trovi nella directory di lavoro corrente del client ftp.**	
Nota 3	**Un messaggio informa che TwidoPort viene riavviato in modo automatico dopo 5 secondi dall'esecuzione corretta del collegamento ftp.**	

Aggiornamento del firmware in corso

La figura seguente mostra una tipica schermata con il messaggio che indica che **Firmware Upgrade In-Progress**:

```
        Telemecanique 499 TWD 01 100 Configuration and Diagnostics
        ---------------------------------------------------

        FIRMWARE UPGRADE IN-PROGRESS...
        Module will reboot in 5 Seconds.
        After Reboot, Connect via FTP to download new Firmware.

        FTP Instructions:
                1) Connect via FTP:  ftp  192.168.2.160
                2) Change to /fw directory:  ftp>cd fw
                3) Download new fw:  ftp>put App.out

        After the FTP download is complete, the module will reboot automatically
.

        Rebooting now.  Goodbye.


Connection to host lost.
```

Modalità Kernel

In mancanza di un firmware valido, TwidoPort passa alla modalità **Kernel**. Se si tenta di utilizzare Telnet per collegarsi a TwidoPort mentre è attiva questa modalità, viene visualizzato il seguente messaggio:

```
Telemecanique 499 TWD 01 100
-----------------------------------------------------
Kernel Version 90.02d

Download valid Exec,App.out, to leave kernel mode.

   To exit type 'quit' 'QUIT' or control D
```

Se si dimentica la password e/o la configurazione IP

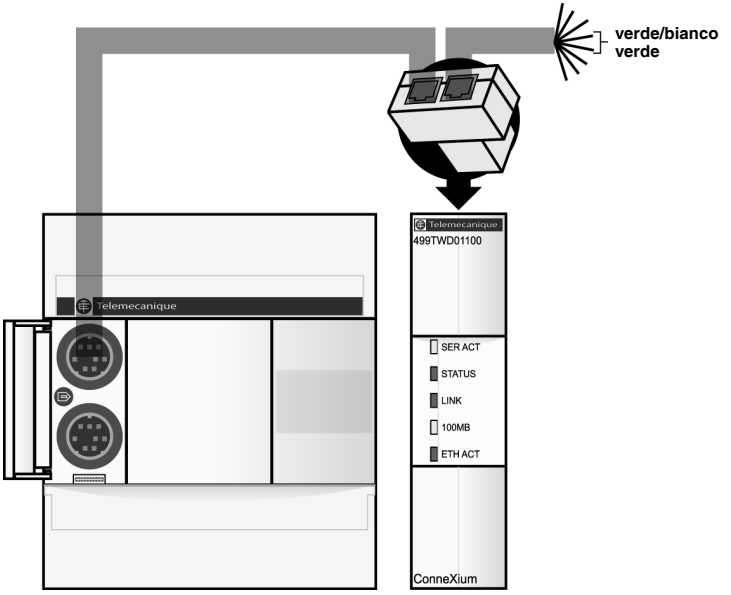
Connessione in modalità backup

Per collegarsi a TwidoPort in modalità backup, procedere come segue:

Passo	Azione	Commento
1	Collegare il pin 3 al pin 6 (terra) del connettore seriale.	Utilizzare il connettore a T RJ-45 Schneider 170 XTS 04 100. *(Vedere l'illustrazione seguente).*
2	Collegarsi a TwidoPort tramite ftp. *(Vedere nota).*	TwidoPort utilizza la seguente configurazione IP predefinita: ● Indirizzo IP: 192.168.2.102 ● Subnet mask: 255.255.0.0 ● Indirizzo gateway: 192.168.2.102 ● Tipo di frame: Ethernet II
3	Aprire il file **fw/Conf.dat.**	Ricavare la configurazione IP e la password dal file **Conf.dat**.
4	Aprire il file **Conf.dat** con un editor di testo.	
Nota	**Non è richiesta alcuna password.**	

Connessione FTP

L'illustrazione seguente mostra come eseguire il collegamento a TwidoPort tramite ftp in modalità backup:



11.3 Funzioni di comunicazione

In breve

Contenuto della sezione

Questa sezione descrive le funzioni di comunicazione supportate dal gateway Ethernet ConneXium TwidoPort.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Funzioni Ethernet	305
Protocollo di comunicazione Modbus/TCP	306
Codici funzione Modbus supportati localmente	307

Funzioni Ethernet

Introduzione

TwidoPort ConneXium aggiunge la connettività Ethernet alla linea di prodotto Twido di Telemecanique. Questo modulo rappresenta il gateway tra un singolo dispositivo Twido Modbus/RTU (RS-485) e il livello fisico delle reti Modbus/TCP in modalità slave. TwidoPort non richiede alimentazione separata, perché viene alimentato dal controller Twido tramite la porta seriale. Questo modulo gateway supporta solo la modalità slave.

Funzioni Ethernet

TwidoPort supporta le seguenti funzioni Ethernet:

- **Negoziazione automatica**
  TwidoPort supporta la negoziazione automatica 10/100TX e comunica solo in modalità half-duplex.
- **Auto-MDI/MDI-X**
  TwidoPort supporta la commutazione automatica delle coppie di fili di trasmissione e ricezione per le comunicazioni con il dispositivo finale (auto-MDI/MDI-X). TwidoPort si connette quindi in modo trasparente l'infrastruttura e i dispositivi finali con cavi diretti o incrociati.

Protocollo di comunicazione Modbus/TCP

Informazioni su Modbus

Il protocollo Modbus è un protocollo di tipo "master/slave" che permette a un master di richiedere risposte dagli slave o di eseguire azioni basate sulle richieste di questi ultimi. Il master può interrogare ogni singolo slave o può inizializzare un messaggio da trasmettere broadcast a tutti gli slave. Gli slave rispondono con un messaggio (risposta) alle richieste che vengono inviate loro singolarmente. Non vengono rinviate le risposte alle query provenienti da un master.

Informazioni sulle comunicazioni Modbus/TCP

TwidoPort supporta fino a otto connessioni Modbus/TCP contemporaneamente. Se si tenta di utilizzare più di otto connessioni si riducono le prestazioni, perché TwidoPort chiude le connessioni che presentano il tempo di inattività più lungo per accettare le nuove richieste.

Principi teorici del funzionamento

I client Modbus/TCP possono comunicare con Twido mediante TwidoPort, tramite un ponte tra dispositivi Twido (Modbus/RTU su collegamento seriale RS-485) e tramite Modbus/TCP su reti Ethernet.

Nota: quando si implementa TwidoPort su una rete, i requisiti di progettazione del sistema devono tenere in conto il limite intrinseco della larghezza di banda delle connessioni seriali. Si otterrà una prestazione di picco di circa 40 transazioni Modbus al secondo. La richiesta di più registri in un'unica richiesta è più efficiente dell'invio di una richiesta separata per ogni registro.

Non è possibile avviare richieste di lettura o scrittura dal controller Twido tramite TwidoPort.

Codici funzione Modbus supportati localmente

Codici funzione List

TwidoPort risponde ai seguenti codici funzione Modbus supportati localmente solo quando l'ID unità è impostato a 254. Ai codici funzione supportati localmente risponde direttamente TwidoPort e non il controller Twido.

Codice funzione Modbus	Codice sottofunzione	OPCODE	Descrizione
8	0	N/D	Restituzione dati della query
8	10	N/D	Azzeramento contatori
8	11	N/D	Restituzione conteggio messaggi bus
8	12	N/D	Restituzione conteggio degli errori di comunicazione del bus
8	13	N/D	Restituzione conteggio errori di eccezione bus
8	14	N/D	Restituzione conteggio messaggi slave
8	15	N/D	Restituzione conteggio assenza di risposta dagli slave
8	21	3	Richiesta statistiche Ethernet
8	21	4	Cancellazione statistiche Ethernet
43	14	N/D	Lettura ID dispositivo (vedere *nota 1*).
Nota 1	**TwidoPort supporta solo gli ID degli oggetti di base del codice funzione di identificazione dei dispositivi di lettura con accesso stream e individuale.**		

Nota: per dettagli sui formati dei messaggi e le classi di accesso, vedere la specifica Modbus all'indirizzo *www.modbus.org*.

Funzionamento del visualizzatore

12

In breve

Oggetto di questo capitolo

Questo capitolo fornisce informazioni dettagliate sul pannello visualizzatore Twido opzionale.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Pannello visualizzatore	310
Informazioni di identificazione e di stato del controller	313
Variabili e oggetti di sistema	315
Impostazioni delle porte seriali	322
Orologio in tempo reale	323
Fattore di correzione dell'orologio-datario	324

Pannello visualizzatore

Introduzione

Il pannello visualizzatore è un'opzione Twido che permette la visualizzazione e il controllo dei dati dell'applicazione e alcune funzioni del controller, come ad esempio lo stato operativo e l'orologio in tempo reale (RTC). Questa opzione è disponibile come cartuccia (TWDXCPODC) per i controller compatti o come modulo di espansione (TWDXCPODM) per i controller modulari.
Il pannello visualizzatore ha due modalità operative:

- Modalità di visualizzazione: visualizza solo i dati.
- Modalità di modifica: consente di modificare i dati.

Nota: il pannello visualizzatore viene aggiornato a un intervallo specifico del ciclo di scansione del controller. Questo può confondere nell'interpretazione della visualizzazione di uscite dedicate per gli impulsi %PLS o %PWM. Quando queste uscite vengono campionate, il loro valore sarà sempre zero e questo valore viene visualizzato.

Visualizzazioni e funzioni

Il pannello visualizzatore fornisce le seguenti visualizzazioni separate con le funzioni associate che si possono eseguire per ciascuna visualizzazione.

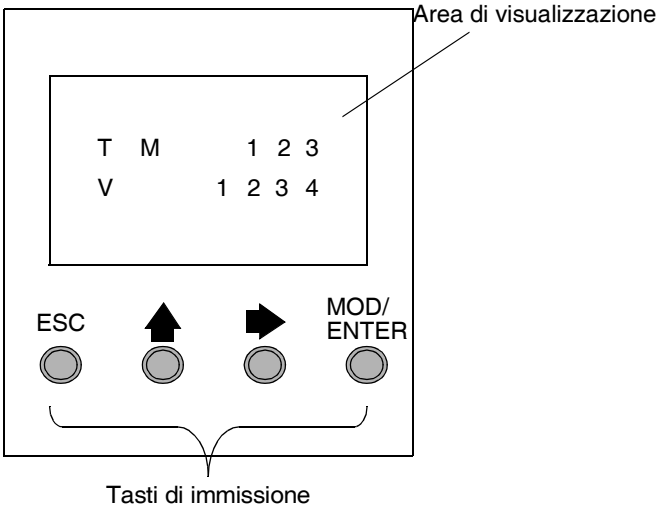
- Identificazione del controller e informazioni di stato: Pannello visualizzatore
  Visualizzazione della revisione del firmware e stato del controller. Cambiare lo stato del controller con i comandi Run, Initial e Stop.
- Oggetti e variabili di sistema: Visualizzazione dati
  Selezionare i dati dell'applicazione in base all'indirizzo: %I, %Q e tutti gli altri oggetti software sul controller di base. Controllo e modifica del valore di un oggetto di dati software selezionato.
- Impostazioni porta seriale: Visualizzazione comunicazioni
  Visualizzazione e modifica delle impostazioni delle porte di comunicazione.
- Ora datario: Visualizzazione data e ora
  Visualizzazione e configurazione della data e dell'ora correnti (se l'RTC è installato).
- Correzione Real time: Fattore RTC:
  Visualizzazione e modifica del valore di correzione RTC per l'RTC opzionale.

Nota:

1. I controller compatti della serie TWDLCA•40DRF integrano la cartuccia RTC.
2. Su tutti gli altri controller, l'orologio in tempo reale e la funzione di correzione in tempo reale sono disponibili solo se è installata la cartuccia opzionale RTC (TWDXCPRTC).

Illustrazione

L'illustrazione seguente mostra una vista del pannello visualizzatore, costituito da un'area di visualizzazione (qui riportata in modalità normale) e quattro tasti di immissione.



Area di visualizzazione

Il pannello visualizzatore include un display LCD in grado di visualizzare due righe di caratteri:

- La prima riga del visualizzatore ha 3 caratteri a 13 segmenti e 4 caratteri a 7 segmenti.
- La seconda riga ha un carattere a 13 segmenti, un carattere a 3 segmenti (per il segno più/meno) e 5 caratteri a 7 segmenti.

Nota: in modalità normale, la prima riga indica il nome dell'oggetto e la seconda il suo valore. In modalità dati, la prima riga visualizza il valore di %SW68 e la seconda il valore di %SW69.

Tasti di immissione

Le funzioni dei quattro tasti di immissione dipendono dalla modalità del pannello visualizzatore:

Tasto	In modalità di visualizzazione	In modalità di modifica
ESC		Ignora le modifiche e torna alla visualizzazione precedente.
▲		Passa al valore successivo di un oggetto in fase di modifica.
➡	Passa alla visualizzazione successiva.	Passa al successivo tipo di oggetto da modificare.
MOD/ ENTER	Entra in modalità di modifica.	Accetta le modifiche e torna alla visualizzazione precedente.

Selezione e spostamento nelle visualizzazioni

La visualizzazione o schermata iniziale del pannello visualizzatore mostra l'identificazione del controller e le informazioni di stato. Premere il tasto ➡ per spostarsi sequenzialmente tra le varie visualizzazioni. Le schermate dell'orologio in tempo reale o del fattore di correzione in tempo reale non sono visualizzate se la cartuccia RTC opzionale (TWDXCPRTC) non viene rilevata sul controller.
Come tasto di scelta rapida, premere il tasto ESC per tornare alla schermata iniziale. Per la maggior parte delle schermate, premendo ESC si ritorna alla schermata di identificazione del controller e delle informazioni di stato. Solo quando si modifica un oggetto e una variabile di sistema che non è una voce iniziale (%I0.0.0), premendo ESC si ritorna alla voce oggetto di sistema prima o iniziale.

Per modificare il valore di un oggetto, anziché premere il tasto ➡ per andare alla prima cifra del valore, premere nuovamente MOD/ENTER.

Informazioni di identificazione e di stato del controller

Introduzione

La visualizzazione o la schermata iniziale del pannello visualizzatore opzionale Twido mostra informazioni sull'identificazione e sullo stato del controller.

Esempio

La revisione del firmware è visualizzata nell'angolo superiore destro dell'area di visualizzazione e lo stato del controller è visualizzato nell'angolo superiore destro dell'area di visualizzazione, nel seguente modo:



Stati del controller

Gli stati del controller includono:

- **NCF: Non configurato**
  Il controller è in uno stato NCF finché non viene caricata un'applicazione. Non è ammesso alcun altro stato finché non viene caricato il programma applicativo. È possibile provare l'I/O modificando il bit di sistema S8 (vedere *Bit di sistema (%S), p. 604*).
- **STP: Applicazione interrotta**
  Se un'applicazione è presente sul controller, lo stato cambia in STP o Stopped. In questo stato, l'applicazione non viene eseguita. Gli ingressi vengono aggiornati e i valori dei dati sono mantenuti sull'ultimo valore. In questo stato, le uscite non sono aggiornate.
- **INI: Iniziale**
  È possibile scegliere di passare allo stato INI o iniziale del controller solo dallo stato STP. L'applicazione non viene eseguita. Gli ingressi del controller sono aggiornati e i valori dei dati sono impostati allo stato iniziale. In questo stato, le uscite non sono aggiornate.
- **RUN: In esecuzione**
  Nello stato RUN l'applicazione è in esecuzione. Gli ingressi del controller sono aggiornati e i valori dei dati sono impostati in base all'applicazione. Questo è l'unico stato in cui le uscite sono aggiornate.
- **HLT: Interrotto (errore dell'applicazione utente)**
  Se il controller è entrato in uno stato ERR o di errore, l'applicazione viene bloccata. Gli ingressi vengono aggiornati e i valori dei dati sono mantenuti sull'ultimo valore. In questo stato le uscite non vengono aggiornate. Il codice di errore viene visualizzato nella parte inferiore destra del pannello visualizzatore come decimale senza segno.
- **NEX: Not Executable (Applicazione non eseguibile)**
  Alla logica utente è stata apportata una modifica in linea. Conseguenza: l'applicazione non è più eseguibile. Ritornerà eseguibile solo dopo che tutte le cause che hanno provocato lo stato Non Exec saranno risolte.

Visualizzazione e modifica degli stati del controller

Utilizzando il pannello visualizzatore, è possibile passare allo stato INI dallo stato STP o da STP a RUN o da RUN a STP. Per cambiare lo stato del controller, procedere come segue:

Passo	Azione
1	Premere il tasto ➡ finché non compare il visualizzatore. In alternativa, premere il tasto ESC. Gli stati correnti del controller sono visualizzati nell'angolo superiore sinistro dell'area di visualizzazione.
2	Premere il tasto MOD/ENTER per entrare nella modalità di modifica.
3	Premere il tasto ⬆ per selezionare uno stato del controller.
4	Premere il tasto MOD/ENTER per accettare il valore modificato o ESC per ignorare le variazioni apportate in modalità di modifica.

Variabili e oggetti di sistema

Introduzione

Il pannello visualizzatore opzionale fornisce le seguenti caratteristiche per il controllo e la regolazione dei dati dell'applicazione:

- selezione dei dati dell'applicazione in base all'indirizzo (ad esempio %I o %Q);
- controllo del valore di un oggetto/variabile software selezionato;
- modifica del valore dell'oggetto dati correntemente visualizzato (inclusa la forzatura di ingressi e uscite).

Oggetti e variabili di sistema

La tabella seguente elenca le variabili e gli oggetti di sistema, nell'ordine di accesso, che possono essere visualizzati e modificati tramite il pannello visualizzatore.

Oggetto	Variabile/Attributo	Descrizione	Accesso
Ingresso	%Ix.y.z	Valore	Lettura/Forzatura
Uscita	%Qx.y.z	Valore	Lettura/Scrittura/Forzatura
Timer	%TMX.V %TMX.P %TMX.Q	Valore corrente Valore preimpostato Completato	Lettura/Scrittura Lettura/Scrittura Lettura
Contatore	%Cx.V %Cx.P %Cx.D %Cx.E %Cx.F	Valore corrente Valore preimpostato Completato Vuoto Pieno	Lettura/Scrittura Lettura/Scrittura Lettura Lettura Lettura
Bit di memoria	%Mx	Valore	Lettura/Scrittura
Parola di memoria	%MWx	Valore	Lettura/Scrittura
Costante	%KWx	Valore	Lettura
Bit di sistema	%Sx	Valore	Lettura/Scrittura
Parola di sistema	%SWx	Valore	Lettura/Scrittura
Ingresso analogico	%IWx.y.z	Valore	Lettura
Uscita analogica	%QWx.y.z	Valore	Lettura/Scrittura
Contatore veloce (FC)	%FCx.V %FCx.VD* %FCx.P %FCx.PD* %FCx.D	Valore corrente Valore corrente Valore preimpostato Valore preimpostato Completato	Lettura Lettura Lettura/Scrittura Lettura/Scrittura Lettura

Oggetto	Variabile/Attributo	Descrizione	Accesso
Contatore veloce (VFC)	%VFCx.V %VFCx.VD* %VFCx.P %VFCx.PD* %VFCx.U %VFCx.C %VFCx.CD* %VFCx.S0 %VFCx.S0D* %VFCx.S1 %VFCx.S1D* %VFCx.F %VFCx.T %VFCx.R %VFCx.S	Valore corrente Valore corrente Valore preimpostato Valore preimpostato Direzione conteggio Valore di cattura Valore di cattura Valore soglia 0 Valore soglia 0 Valore soglia 1 Valore soglia 1 Overflow Base tempo Uscita riflessa abilitata Ingresso riflesso abilitato	Lettura Lettura Lettura/Scrittura Lettura/Scrittura Lettura Lettura Lettura Lettura/Scrittura Lettura/Scrittura Lettura/Scrittura Lettura/Scrittura Lettura Lettura/Scrittura Lettura/Scrittura Lettura/Scrittura
Ingresso parola di rete	%INWx.z	Valore	Lettura
Uscita parola di rete	%QNWx.z	Valore	Lettura/Scrittura
Grafcet	%Xx	Bit di passo	Lettura
Generatore d'impulsi	%PLS.N %PLS.ND* %PLS.P %PLS.D %PLS.Q	Numero di impulsi Numero di impulsi Valore preimpostato Completato Uscita di corrente	Lettura/Scrittura Lettura/Scrittura Lettura/Scrittura Lettura Lettura
Modulatore d'ampiezza d'impulso	%PWM.R %PWM.P	Rapporto Valore preimpostato	Lettura/Scrittura Lettura/Scrittura
Controller a tamburo	%DRx.S %DRx.F	Numero passo corrente pieno	Lettura Lettura
Contatore passo	%SCx.n	Bit contatore di passo	Lettura/Scrittura
Registro	%Rx.I %Rx.O %Rx.E %Rx.F	Ingresso Uscita Vuoto Pieno	Lettura/Scrittura Lettura/Scrittura Lettura Lettura
Registro bit di scorrimento	%SBR.x.yy	Bit di registro	Lettura/Scrittura
Messaggio	%MSGx.D %MSGx.E	Completato errore	Lettura Lettura
Ingresso slave AS-Interface	%IAx.y.z	Valore	Lettura/Forzatura
Ingresso analogico slave AS-Interface	%IWAx.y.z	Valore	Lettura
Uscita slave AS-Interface	%QAx.y.z	Valore	Lettura/Scrittura/Forzatura
Uscita analogica slave AS-Interface	%QWAx.y.z	Valore	Lettura/Scrittura
Ingresso PDO slave CANopen	%IWCx.y.z	Valore parola singola	Lettura
Uscita PDO slave CANopen	%QWCx.y.z	Valore parola singola	Lettura/Scrittura

Note:

1. (*)significa una variabile di parola doppia a 32 bit. L'opzione parola doppia è disponibile su tutti i controller ad eccezione dei controller Twido TWDLC•A10DRF.
2. Le variabili non saranno visualizzate se non sono utilizzate in un'applicazione, poiché Twido utilizza l'assegnazione dinamica della memoria.
3. Se il valore di %MW è maggiore di +32767 o inferiore a -32768, il pannello visualizzatore continuerà a lampeggiare.
4. Se il valore di %SW è maggiore di 65535, il pannello visualizzatore continua a lampeggiare, eccetto che per %SW0 e %SW11. Se si immette un valore maggiore del limite, esso tornerà al valore configurato.
5. Se il valore immesso per %PLS.P è superiore al limite, il valore scritto sarà quello di saturazione.

Visualizzazione e modifica di oggetti e variabili

È possibile accedere a ciascun tipo di oggetto di sistema partendo con l'oggetto ingresso (%I), passando in sequenza all'oggetto messaggio (%MSG) e infine ritornando all'oggetto ingresso (%I).
Per visualizzare un oggetto di sistema:

Passo	Azione
1	Premere il tasto ➡ finché non compare il visualizzatore dei dati. L'oggetto ingresso ("I") sarà visualizzato nell'angolo superiore sinistro dell'area di visualizzazione. Il carattere "I" (o il nome dell'oggetto visualizzato in precedenza come dati) non lampeggia.
2	Premere il tasto MOD/ENTER per entrare nella modalità di modifica. Il carattere "I" dell'oggetto ingresso (o il nome dell'oggetto visualizzato in precedenza come dati) inizia a lampeggiare.
3	Premere il tasto ⬆ per passare in sequenza l'elenco di oggetti.
4	Premere il tasto ➡ per passare in sequenza il campo di un tipo di oggetto e premere il tasto ⬆ per incrementare il valore di tale campo. È possibile utilizzare il tasto ➡ e il tasto ⬆ per spostarsi e modificare tutti i campi dell'oggetto visualizzato.
5	Ripetere i passi 3 e 4 finché la modifica non è completa.
6	Premere il tasto MOD/ENTER per accettare i valori modificati. Nota: il nome e l'indirizzo dell'oggetto devono essere convalidati prima di accettare qualsiasi modifica, ossia devono esistere nella configurazione del controller prima di utilizzare il pannello visualizzatore. Premere ESC per ignorare le variazioni apportate in modalità di modifica.

Valori di dati e formati di visualizzazioni

In generale, il valore dati per un oggetto o una variabile viene mostrato come intero con o senza segno nella parte inferiore destra dell'area di visualizzazione. Inoltre, tutti i campi eliminano gli zeri iniziali per i valori visualizzati. L'indirizzo di ciascun oggetto viene visualizzato nel pannello visualizzatore in uno dei seguenti sette formati:

- Formato I/O
- Formato I/O degli slave AS-Interface
- Formato I/O slave CANopen
- Formato blocco funzione
- Formato semplice
- Formato I/O di rete
- Formato contatore di passo
- Formato registro bit di scorrimento

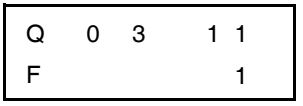
Formato ingresso/uscita

Gli oggetti ingresso/uscita (%I, %Q, %IW e %QW) hanno indirizzi a tre parti, come ad esempio %IX.Y.Z e sono visualizzati nel seguente modo:

- Tipo di oggetto e indirizzo del controller in alto a sinistra
- Indirizzo dell'espansione nella parte centrale superiore
- Canale di I/O in alto a destra

Nel caso di un ingresso (%I) e di un'uscita (%Q) semplici, la parte in basso a sinistra della visualizzazione contiene il carattere "U" (unforced) per un bit non forzato o "F" (forced) per indicare un bit forzato. Il valore di forzatura viene visualizzato in basso a destra nella schermata.

L'oggetto uscita %Q0.3.11 compare nell'area di visualizzazione nel seguente modo:



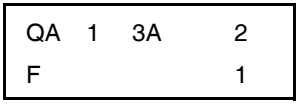
Formato degli I/O slave AS-Interface

Gli oggetti ingresso/uscita degli slave AS-Interface (%IA, %QA, %IWA e %QWA) hanno indirizzi in quattro parti, come ad esempio %IAx.y.z e sono visualizzati nel seguente modo:

- Tipo dell'oggetto in alto a sinistra
- Indirizzo del master AS-Interface sul bus di espansione nella parte superiore centrale sinistra
- Indirizzo dello slave sul bus AS-Interface nella parte superiore centrale destra
- Canale di I/O dello slave nella parte superiore destra.

Nel caso di un ingresso (%IA) e di un'uscita (%QA) semplici, in basso a sinistra viene visualizzato il carattere "U" (unforced) per un bit non forzato o "F" (forced) per indicare un bit forzato. Il valore di forzatura viene visualizzato in basso a destra nella schermata.

L'oggetto uscita %QA1.3A.2 appare nell'area di visualizzazione nel seguente modo:

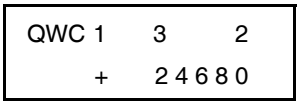


Formato I/O slave CANopen

Gli oggetti I/O PDO degli slave CANopen (%IWC e %QWC) hanno indirizzi costituiti da quattro parti, come ad esempio %IWCx.y.z e sono visualizzati nel seguente modo:

- Tipo dell'oggetto in alto a sinistra
- Indirizzo del master CANopen sul bus di espansione nella parte superiore centrale sinistra
- Indirizzo dello slave sul bus CANopen nella parte superiore centrale destra
- Canale di I/O PDO dello slave nella parte superiore destra.
- Valore con segno dell'oggetto in basso.

Nell'esempio seguente, l'oggetto uscita PDO %QWC1.3.2 contiene il valore con segno +24680:

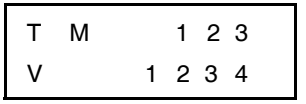


Formato blocco funzione

I blocchi funzione (%TM, %C, %FC, %VFC, %PLS, %PWM, %DR, %R e %MSGj) hanno indirizzi in due parti che contengono un numero di oggetto e un nome di variabile o di attributo e sono visualizzati nel seguente modo:

- Nome del blocco funzione in alto a sinistra
- Numero (o istanza) del blocco funzione in alto a destra
- Variabile o attributo in basso a sinistra
- Valore dell'attributo in basso a destra.

Nell'esempio che segue, il valore corrente per il timer numero 123 è impostato a 1.234.

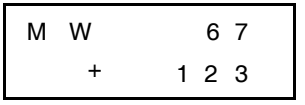


Formato semplice

Per gli oggetti %M, %MW, %KW, %MD, %KD, %MF, %KF, %S, %SW e %X viene utilizzato un formato semplice:

- Numero di oggetto in alto a destra.
- Valore con segno per gli oggetti in basso.

Nell'esempio che segue, la parola di memoria numero 67 contiene il valore +123.

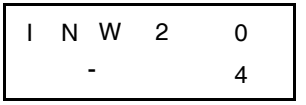


Formato I/O di rete

Gli oggetti I/O di rete (%INW e %QNW) appaiono nell'area di visualizzazione nel seguente modo:

- Nome dell'oggetto in alto a sinistra
- Indirizzo del controller in alto al centro.
- Numero di oggetto in alto a destra.
- Valore con segno dell'oggetto in basso.

Nell'esempio che segue, la prima parola d'ingresso di rete del controller remoto configurato come indirizzo remoto 2 è impostata al valore -4.

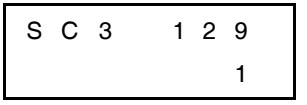


Formato contatore di passo

Il formato contatore di passo (%SC) visualizza il numero dell'oggetto e il bit contatore di passo nel seguente modo:

- Nome e numero dell'oggetto in alto a sinistra.
- Numero del bit contatore di passo in alto a destra.
- Valore bit contatore di passo nella parte inferiore del visualizzatore.

Nell'esempio seguente, il numero bit 129 del contatore di passo numero 3 è impostato a 1.

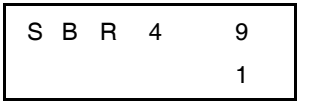


Formato registro bit di scorrimento

Il registro di scorrimento bit (%SBR) appare nell'area di visualizzazione nel seguente modo:

- Nome e numero dell'oggetto in alto a sinistra.
- Numero di bit di registro in alto a destra.
- Valore del bit di registro in basso a destra.

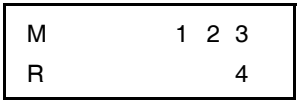
L'esempio che segue mostra la visualizzazione del registro bit di scorrimento numero 4.



Impostazioni delle porte seriali

Introduzione

Il pannello visualizzatore consente di visualizzare le impostazioni del protocollo e di modificare gli indirizzi di tutte le porte seriali mediante TwidoSoft. Il numero massimo di porte seriali è due. Nell'esempio che segue, la prima porta è configurata come protocollo Modbus con indirizzo 123. La seconda è configurata come collegamento remoto con indirizzo 4.



Visualizzazione e modifica delle impostazioni delle porte seriali

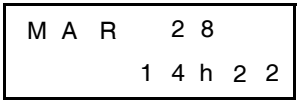
I controller Twido supportano un massimo di due porte seriali. Per visualizzare le impostazioni delle porte seriali utilizzando il pannello visualizzatore:

Passo	Azione
1	Premere il tasto ▶ finché non compare il visualizzatore delle comunicazioni. La lettera singola dell'impostazione del protocollo della prima porta seriale ("M", "R" o "A") comparirà nell'angolo superiore sinistro del pannello visualizzatore.
2	Premere il tasto MOD/ENTER per entrare in modalità di modifica.
3	Premere il tasto ▶ fino a raggiungere il campo che si desidera modificare.
4	Premere il tasto ▲ per incrementare il valore di tale campo.
5	Proseguire con i passi 3 e 4 finché le impostazioni degli indirizzi sono complete.
6	Premere il tasto MOD/ENTER per accettare i valori modificati o ESC per eliminare le variazioni apportate in modalità di modifica.
Nota:l'indirizzo fa parte dei dati di configurazione sul controller. La modifica del suo valore tramite il pannello visualizzatore implica che non si riuscirà più a connettersi mediante TwidoSoft come pari. TwidoSoft richiederà di eseguire lo scaricamento per essere di nuovo pari.	

Orologio in tempo reale

Introduzione

È possibile modificare la data e l'ora con il pannello visualizzatore se sul controller Twido in uso è installata la cartuccia opzionale RTC (TWDXCPRTC). Il mese viene visualizzato sul lato superiore sinistro del visualizzatore HMI. Finché non viene immesso il valore esatto dell'ora, il campo mese conterrà la scritta "RTC". Il giorno del mese è visualizzato nell'angolo superiore destro del visualizzatore. L'ora è in formato militare. Ore e minuti sono indicati nell'angolo inferiore destro del visualizzatore e sono separati dalla lettera "h". L'esempio che segue mostra che l'RTC è impostato su 28 marzo, 2:22 PM.



Nota:
1. I controller compatti della serie TWDLCA•40DRF integrano la cartuccia RTC.
2. Su tutti gli altri controller, l'orologio in tempo reale e la funzione di correzione sono disponibili solo se è installata la cartuccia opzionale RTC (TWDXCPRTC).

Visualizzazione e regolazione dell'orologio in tempo reale

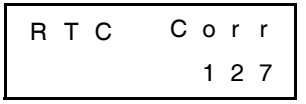
Per visualizzare e regolare l'orologio in tempo reale:

Passo	Azione
1	Premere il tasto ▶ finché non viene visualizzata la data e l'ora. Il valore del mese ("GEN", "FEB") comparirà nell'angolo superiore sinistro dell'area di visualizzazione. Se non è stato inizializzato alcun mese, viene visualizzata la scritta "RTC" nell'angolo superiore sinistro.
2	Premere il tasto MOD/ENTER per entrare in modalità di modifica.
3	Premere il tasto ▶ fino a raggiungere il campo che si desidera modificare.
4	Premere il tasto ▲ per incrementare il valore di tale campo.
5	Continuare con i passi 3 e 4 per completare la regolazione.
6	Premere il tasto MOD/ENTER per accettare i valori modificati o ESC per ignorare le modifiche apportate in modalità di modifica.

Fattore di correzione dell'orologio-datario

Introduzione

È possibile visualizzare e modificare il fattore di correzione dell'orologio-datario (RTC) utilizzando il pannello visualizzatore. Ogni modulo opzionale orologio-datario dispone di un valore del fattore di correzione dell'RTC utilizzato per correggere le imprecisioni del display a cristalli liquidi del modulo. Il fattore di correzione è un numero intero a 3 cifre senza segno compreso tra 0 e 127. Questo valore è visualizzato nell'angolo inferiore destro del visualizzatore.
Di seguito è riportato un esempio di un fattore di correzione di 127.



Visualizzazione e modifica della correzione dell'RTC

Per visualizzare e modificare il fattore di correzione in tempo reale:

Passo	Azione
1	Premere il tasto ➡ finché non appare il visualizzatore del fattore di correzione RTC. Il messaggio "RTC Corr" verrà visualizzato sulla riga superiore del pannello visualizzatore.
2	Premere il tasto MOD/ENTER per entrare nella modalità di modifica.
3	Premere il tasto ➡ fino a raggiungere il campo che si desidera modificare.
4	Premere il tasto ⬆ per incrementare il valore di tale campo.
5	Ripetere i passi 3 e 4 per completare il valore di correzione dell'RTC.
6	Premere il tasto MOD/ENTER per accettare i valori modificati o ESC per ignorare le variazioni apportate in modalità di modifica.

Descrizione dei linguaggi Twido

III

In breve

Contenuto della sezione

Questa sezione fornisce istruzioni per l'utilizzo dei linguaggi di programmazione Ladder, List e Grafcet per creare programmi di controllo per i controller programmabili Twido.

Contenuto di questa parte

Questa parte contiene i seguenti capitoli:

Capitolo	Titolo del capitolo	Pagina
13	Ladder, linguaggio	327
14	Lista d'istruzioni, linguaggio	349
15	Grafcet	361

Ladder, linguaggio

13

In breve

Oggetto di questo capitolo

Questo capitolo descrive la programmazione con il linguaggio Ladder.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Introduzione ai linguaggi Ladder	328
Principi di programmazione dei linguaggi Ladder	330
Blocchi linguaggio Ladder	332
Elementi grafici linguaggio Ladder	335
Istruzioni Ladder speciali OPEN e SHORT	338
Note per la programmazione	339
Reversibilità Ladder/List	343
Indicazioni per la reversibilità Ladder/List	344
Documentazione del programma	346

Introduzione ai linguaggi Ladder

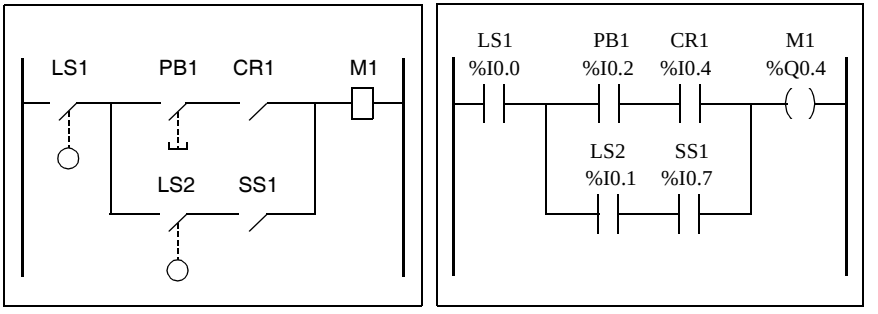
Introduzione

I linguaggi Ladder sono simili ai diagrammi logici di relè che rappresentano i circuiti di controllo dei relè in logica programmata, ma in uno schema di contatti:

- Tutti gli ingressi sono rappresentati da simboli di contatto (⊣⊢).
- Tutte le uscite sono rappresentate da simboli di bobina (⊣( )⊢).
- Le operazioni numeriche sono incluse nella serie di istruzioni grafiche Ladder.

Equivalenti Ladder ai circuiti di relè

L'illustrazione seguente mostra un diagramma di cablaggio semplificato di un circuito logico a relè e il linguaggio Ladder corrispondente.



Circuito logico a relè

Ladder

Tenere presente che nell'illustrazione precedente, tutti gli ingressi associati a un dispositivo di commutazione nel diagramma logico a relè sono indicati come contatti nel linguaggio Ladder. La bobina di uscita M1 nel diagramma logico a relè è rappresentata dal simbolo di bobina di uscita nel linguaggio Ladder. I numeri di indirizzo che appaiono sopra ciascun simbolo contatto/bobina nel linguaggio Ladder fanno riferimento alle ubicazioni delle connessioni esterne di ingresso/uscita verso il controller.

Reti Ladder

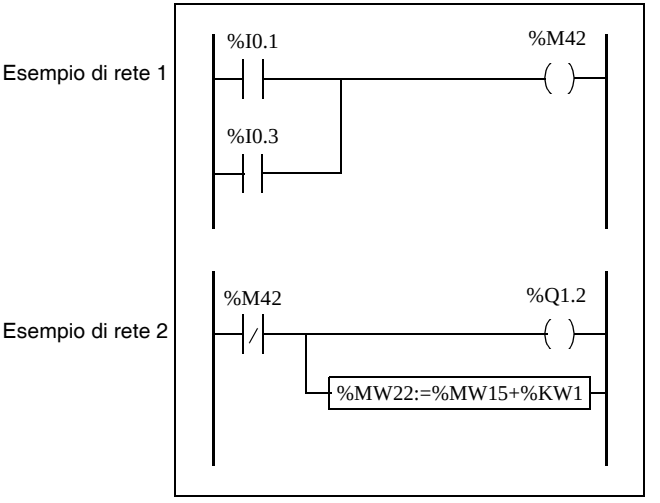
Un programma scritto in linguaggio Ladder è composto da reti che costituiscono insiemi di istruzioni grafiche tra due barre potenziale verticali. Le reti vengono eseguite in sequenza dal controller.
L'insieme di istruzioni grafiche rappresenta le funzioni seguenti:

- Ingressi/uscite del controller (pulsanti, sensori, relè, luci spia,)
- Funzioni del controller (timer, contatori,)
- Operazioni matematiche e logiche (addizione, divisione, AND, XOR,)
- Operatori di confronto e altre operazioni numeriche (A<B, A=B, scorrimento, rotazione e così via)
- Variabili interne nel controller (bit, parole,)

Queste istruzioni grafiche sono disposte con collegamenti verticali e orizzontali che portano a una o più uscite e/o azioni. Una rete non può supportare più di un gruppo di istruzioni collegate.

Esempio di reti Ladder

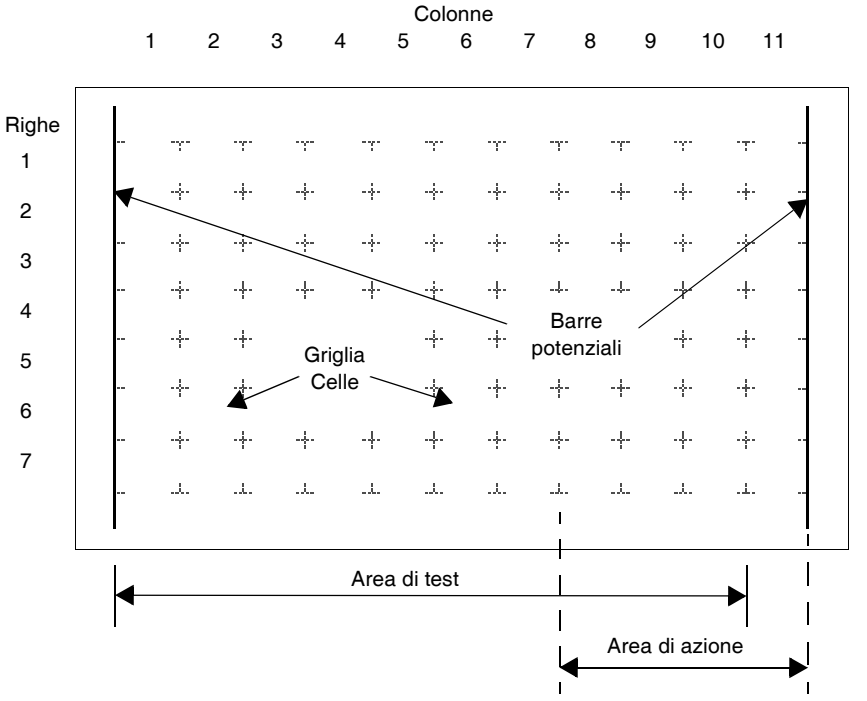
Il diagramma che segue riporta un esempio di programma Ladder composto da due reti.



Principi di programmazione dei linguaggi Ladder

Griglia di programmazione

Ogni rete Ladder consiste in una griglia di sette righe per undici colonne organizzate in due aree come indicato nell'illustrazione che segue.



Aree della griglia

La griglia di programmazione del linguaggio Ladder è divisa in due aree:

- Area di test
  Contiene le condizioni provate per eseguire le azioni. È costituita dalle colonne da 1 a 10 e contiene contatti, blocchi funzione e blocchi di confronto.
- Area di azione
  Contiene l'uscita o l'operazione che verrà eseguita in base ai risultati dei test delle condizioni contenuti nell'area di test. È costituita dalle colonne da 8 a 11 e contiene bobine e blocchi di operazioni.

Inserimento delle istruzioni nella griglia

La rete Ladder presenta una griglia di programmazione sette per undici che inizia nella prima cella nell'angolo in alto a sinistra della griglia. La programmazione consiste nell'inserimento delle istruzioni nelle celle della griglia. Le istruzioni di test, i confronti e le funzioni vengono inserite nelle celle nell'area di test e sono allineate a sinistra. La logica di test fornisce continuità con l'area di azione in cui si inseriscono bobine, operazioni numeriche e istruzioni di controllo del flusso di programma, allineate a destra.
La rete viene risolta o eseguita (test effettuati e uscite assegnate) all'interno della griglia dall'alto verso il basso e da sinistra a destra.

Intestazioni della rete

Oltre alla rete, viene visualizzata in alto anche un'intestazione. Utilizzare l'intestazione per documentare lo scopo logico della rete. L'intestazione della rete può contenere le informazioni seguenti:

- Numero di rete
- Etichette (%Li)
- Dichiarazioni di subroutine (SRi:)
- Titolo della rete
- Commenti alla rete

Per ulteriori informazioni sull'utilizzo dell'intestazione di rete per documentare i programmi, consultare *Documentazione del programma, p. 346*.

Blocchi linguaggio Ladder

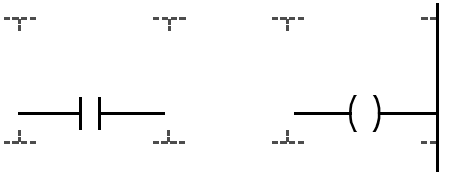
Introduzione

I linguaggi Ladder sono costituiti da blocchi che rappresentano le funzioni e il flusso di programma, come indicato di seguito:

- Contatti
- Bobine
- Istruzioni di flusso programma
- Blocchi funzione
- Blocchi confronto
- Blocchi operate

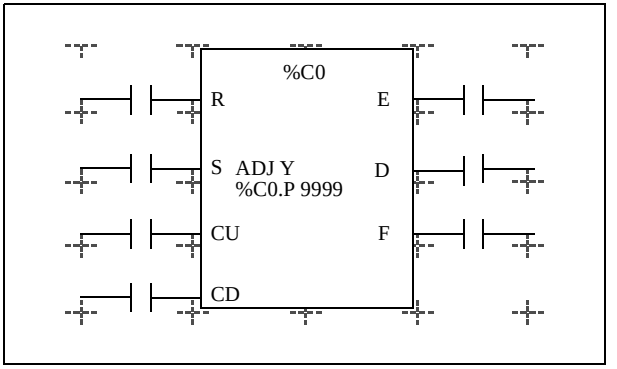
Contatti, bobine e flusso programma

I contatti, le bobine e le istruzioni di flusso programma (salto e chiamata) occupano una singola cella della griglia di programmazione Ladder. I blocchi funzione, confronto e operate occupano più celle.
Di seguito vengono forniti esempi di contatto e bobina.



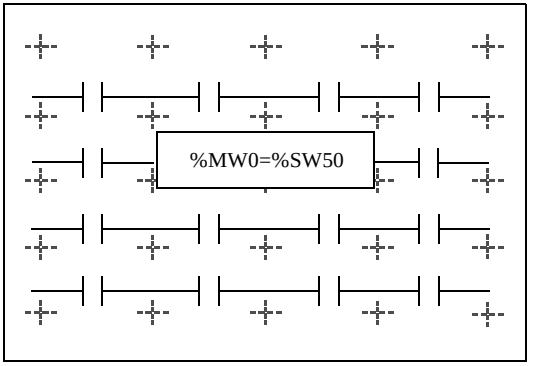
Blocchi funzione

I blocchi funzione vengono posizionati nell'area di test della griglia di programmazione. I blocchi devono apparire nella prima riga; sopra o sotto questi blocchi non devono essere presenti istruzioni Ladder o linee di continuità. Le istruzioni di test Ladder portano al lato ingresso del blocco funzione e le istruzioni di test e/o di azione portano al lato uscita del blocco.
I blocchi funzione sono orientati verticalmente e occupano due colonne per quattro righe della griglia di programmazione.
Di seguito è riportato un esempio di un blocco funzione contatore.



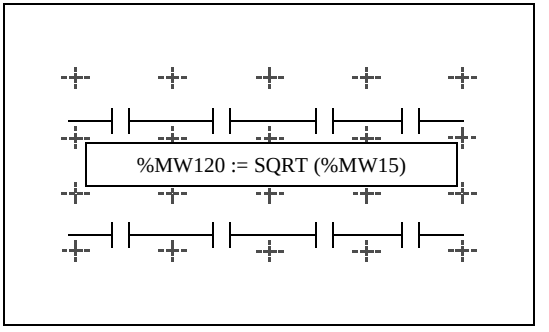
Blocchi confronto

I blocchi confronto vengono posizionati nell'area di test della griglia di programmazione. Il blocco può trovarsi su una riga o colonna qualsiasi nell'area di test purché l'intera istruzione risieda nell'area di test.
I blocchi confronto sono orientati orizzontalmente e occupano due colonne per una riga della griglia di programmazione.
Vedere l'esempio seguente di blocco confronto.



Blocchi operate

I blocchi operate vengono posizionati nell'area di azione della griglia di programmazione. Il blocco può trovarsi su una riga qualsiasi dell'area di azione. L'istruzione è giustificata a destra, appare sulla destra e termina nell'ultima colonna.
I blocchi operate sono orientati orizzontalmente e occupano quattro colonne per una riga della griglia di programmazione.
Di seguito è riportato un esempio di un blocco operate.



Elementi grafici linguaggio Ladder

Introduzione

Le istruzioni nei diagrammi Ladder sono costituite da elementi grafici.

Contatti

Gli elementi grafici dei contatti sono programmati nell'area di test e occupano una cella (una riga di altezza per una colonna di larghezza).

Nome	Elemento grafico	Istruzione	Funzione
Contatto normalmente aperto		LD	Contatto passante quanto l'oggetto bit di controllo si trova allo stato 1.
Contatto normalmente chiuso		LDN	Contatto passante quanto l'oggetto bit di controllo si trova allo stato 0.
Contatto per individuare un fronte di salita		LDR	Fronte di salita: rilevamento del passaggio da 0 a 1 dell'oggetto bit di controllo.
Contatto per individuare un fronte di discesa		LDF	Fronte di discesa: rilevamento del passaggio da 1 a 0 dell'oggetto bit di controllo.

Elementi di collegamento

Gli elementi grafici di collegamento vengono utilizzati per collegare gli elementi grafici di azione e test.

Nome	Elemento grafico	Funzione
Collegamento orizzontale		Collega in serie gli elementi grafici di azione e test tra due barre di potenziale (alimentazione).
Collegamento verticale		Collega in parallelo gli elementi grafici di test e di azione.

Bobine

Gli elementi grafici bobina sono programmati nell'area di azione e occupano una cella (una riga di altezza per una colonna di larghezza).

Nome	Elemento grafico	Istruzione	Funzione
Bobina diretta		ST	L'oggetto bit associato assume il valore del risultato dell'area di test.
Bobina negata		STN	L'oggetto bit associato assume il valore inverso del risultato dell'area di test.
Imposta bobina		S	L'oggetto bit associato viene impostato a 1 quando il risultato dell'area di test è 1.
Ripristina bobina		R	L'oggetto bit associato viene azzerato quando il risultato dell'area di test è 1.
Salto o chiamata a subroutine	->>%Li ->>%SRi	JMP SR	Collega a una istruzione con label, prima o dopo nel programma.
Bobina condizione di transizione			Linguaggio Grafcet. Utilizzato quando la programmazione delle condizioni di transizione associate alle transizioni provoca l'avanzamento al passo successivo.
Ritorno da una subroutine	<RET>	RET	Posta al termine delle subroutine per tornare al programma principale.
Arresto programma	<END>	END	Definisce il termine del programma.

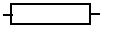
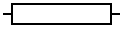
Blocchi funzione

Gli elementi grafici dei blocchi funzione vengono programmati nell'area di test e richiedono un numero di celle pari a quattro righe per due colonne (ad eccezione dei contatori molto veloci che richiedono cinque righe per due colonne).

Nome	Elemento grafico	Funzione
Timer, contatori, registri e così via.		Ciascun blocco funzione utilizza gli ingressi e le uscite che attivano i collegamenti con gli altri elementi grafici. Nota: le uscite dei blocchi funzione non possono essere collegate tra loro (corti verticali).

Blocchi Operate e Confronto

I blocchi confronto vengono programmati nell'area di test e i blocchi operate nell'area di azione.

Nome	Elemento grafico	Funzione
Blocco confronto		Confronta due operandi, l'uscita passa a 1 quando viene controllato il risultato. Dimensione: una riga per due colonne
Blocco operazione		Esegue operazioni aritmetiche e logiche. Dimensione: una riga per quattro colonne

Istruzioni Ladder speciali OPEN e SHORT

Introduzione

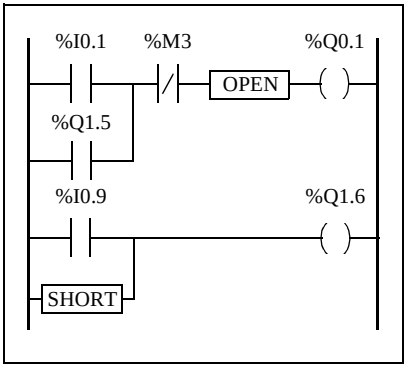
Le istruzioni OPEN e SHORT costituiscono un utile metodo per eseguire il debug e risolvere i problemi dei programmi Ladder. Queste istruzioni speciali alterano la logica di una rete accorciandone o aprendone la continuità come illustrato nella tabella che segue.

Istruzione	Descrizione	Istruzione List
OPEN	Crea un'interruzione nella continuità di una rete Ladder indipendentemente dai risultati dell'ultima operazione logica.	`AND 0`
SHORT	Consente la continuità per il passaggio attraverso la rete indipendentemente dai risultati dell'ultima operazione logica.	`OR 1`

Nella programmazione List, le istruzioni OR e AND vengono utilizzate per creare le istruzioni OPEN e SHORT utilizzando rispettivamente i valori immediati di 0 e 1.

Esempi

Gli esempi che seguono illustrano l'impiego delle istruzioni OPEN e SHORT.



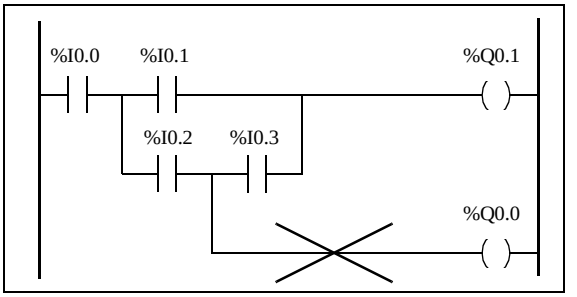

```
LD      %I0.1
OR      %Q1.5
ANDN    %M3
AND     0
ST      %Q0.1
LD      %I0.9
OR      1
ST      %Q1.6
```

Note per la programmazione

Gestione dei salti di programma	Utilizzare i salti di programma con cautela per evitare loop troppo estesi che possono aumentare il tempo di scansione. Evitare salti a istruzioni posizionate a monte. (Una riga di istruzione a monte è posizionata prima di un salto nel programma, mentre una riga di istruzione a valle si trova dopo il salto).
Programmazione delle uscite	I bit di uscita, come i bit interni, possono essere modificati solo una volta nel programma. Nel caso dei bit di uscita, viene considerato solo l'ultimo valore analizzato quando vengono aggiornate le uscite.
Utilizzo dei sensori per l'arresto di emergenza a collegamento diretto	I sensori utilizzati direttamente per gli arresti di emergenza non devono essere trattati dal controller. Essi devono essere collegati direttamente alle uscite corrispondenti.
Gestione del ripristino dell'alimentazione	Condizionare un ritorno di alimentazione a un'operazione manuale. Un riavvio automatico può provocare un funzionamento non desiderato dell'installazione (utilizzare i bit di sistema %S0, %S1 e %S9).
Gestione dell'ora e dei blocchi di programmazione	Controllare lo stato del bit di sistema %S51, che indica eventuali errori dell'orologio-datario.
Controllo di sintassi ed errori	Quando si immette un programma, TwidoSoft controlla la sintassi delle istruzioni, gli operandi e la loro associazione.

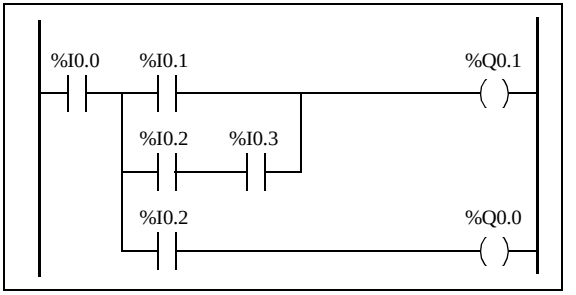
Note aggiuntive relative all'uso delle parentesi

Le operazioni di assegnazione non devono essere incluse tra parentesi.



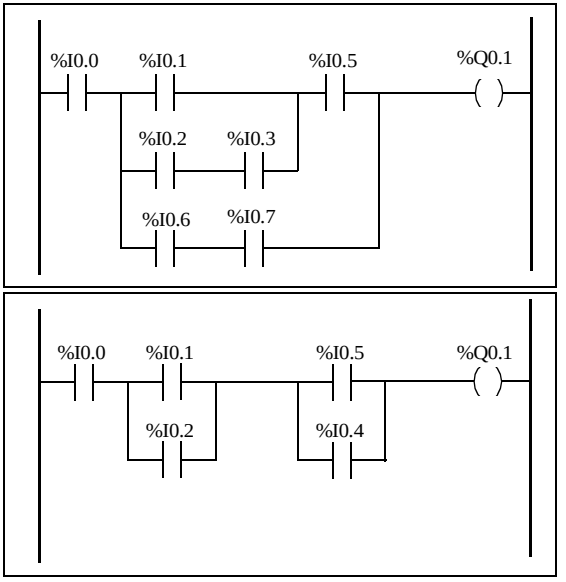

```
LD      %I0.0
AND     %I0.1
OR(     %I0.2
ST      %Q0.0
AND     %I0.3
)
ST      %Q0.1
```

Per eseguire la stessa funzione, occorre programmare le seguenti equazioni:

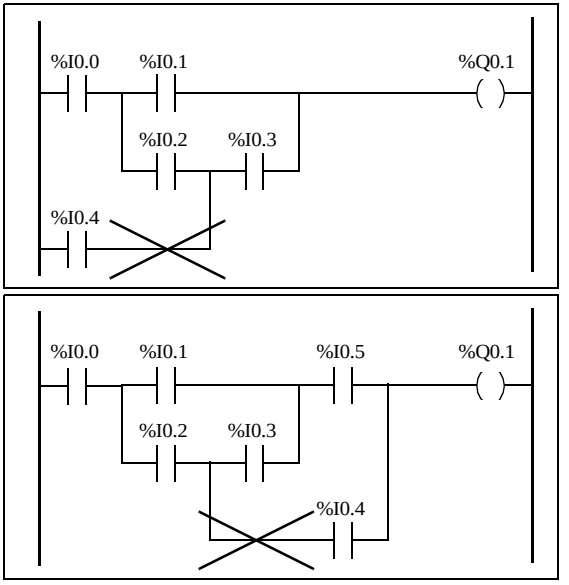



```
LD      %I0.0
MPS
AND(    %I0.1
OR(     %I0.2
AND     %I0.3
)
)
ST      %Q0.1
MPP
AND     %I0.2
```

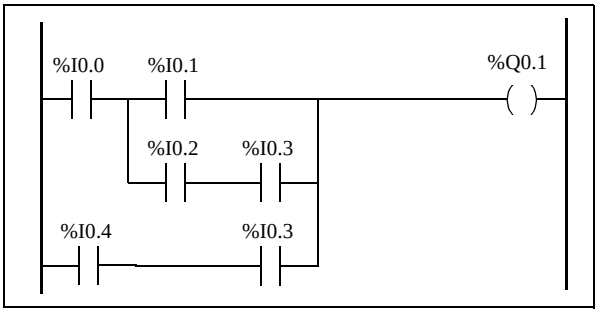
Se si mettono in parallelo più contatti, è necessario annidarli gli uni dentro gli altri oppure dissociarli completamente:



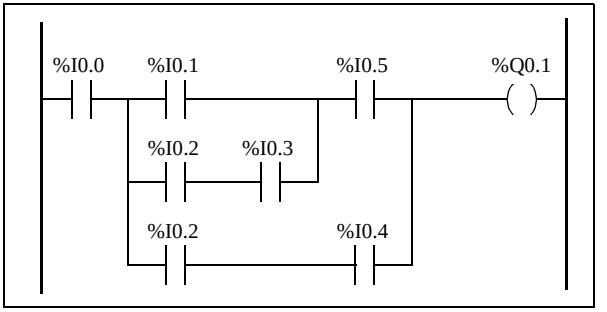
Gli schemi seguenti non possono essere programmati:



Per eseguire schemi equivalenti a quelli indicati, modificarli come segue:




```
LD      %I0.0
AND(    %I0.1
OR(     %I0.2
AND     %I0.3
)
)
OR(     %I0.4
AND     %I0.3
)
ST      %Q0.1
```




```
LD      %I0.0
AND(    %I0.1
OR(     %I0.2
AND     %I0.3
)
AND     %I0.5
OR(     %I0.2
AND     %I0.4
)
)
ST      %Q0.1
```

Reversibilità Ladder/List

Introduzione

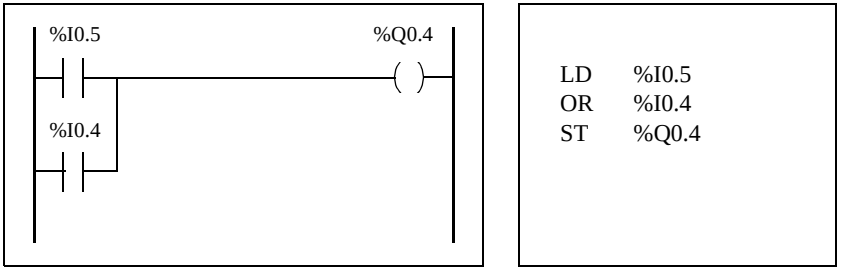
La reversibilità è una funzione del software di programmazione TwidoSoft che consente la conversione dei programmi applicativi da Ladder a List e da List a Ladder.
Le preferenze utente impostate in TwidoSoft permettono di scegliere il metodo di visualizzazione predefinito dei programmi: in formato List o in formato Ladder. TwidoSoft consente inoltre di commutare tra le visualizzazioni List e Ladder.

Come funziona la Reversibilità

Per comprendere la funzione di reversibilità del programma occorre esaminare la relazione di una rete Ladder e la sequenza List di istruzioni associata:

- **Rete Ladder** : insieme di istruzioni Ladder che costituiscono un'espressione logica.
- **Sequenza List** : insieme di istruzioni di un programma List corrispondenti alle istruzioni Ladder e relativo alla medesima espressione logica.

L'illustrazione che segue mostra una rete Ladder normale e il programma logico equivalente espresso come sequenza di istruzioni List.



Un programma di applicativo è memorizzato internamente sotto forma di istruzioni List, indipendentemente dal fatto che sia stato scritto in linguaggio Ladder o List. TwidoSoft utilizza le analogie di programma esistenti tra i due linguaggi e l'immagine List interna del programma per visualizzarlo sia sotto forma di lista di istruzioni (forma elementare), sia in modo grafico, sia sotto la forma Ladder, a seconda delle preferenze selezionate dall'utente.

Come assicurare la Reversibilità

È possibile convertire qualsiasi programma creato in linguaggio Ladder in un programma List. Alcune logiche del linguaggio List, al contrario, non possono essere convertite in linguaggio Ladder. Per garantire una reversibilità totale tra il linguaggio List e il linguaggio Ladder, è importante seguire le indicazioni fornite alla sezione *Indicazioni per la reversibilità Ladder/List, p. 344*.

Indicazioni per la reversibilità Ladder/List

Istruzioni richieste per la reversibilità

La struttura di un blocco funzione reversibile in linguaggio List richiede l'uso delle istruzioni seguenti:

- **BLK** indica l'avvio del blocco, definisce l'inizio della rete e l'avvio della porzione di ingresso nel blocco.
- **OUT_BLK** indica l'inizio della porzione di uscita del blocco.
- **END_BLK** indica la fine del blocco e della rete.

L'utilizzo delle istruzioni di blocco funzione reversibile non è obbligatorio in un programma List correttamente funzionante. Per alcune istruzioni è possibile programmare in List, linguaggio non reversibile. Per una descrizione della programmazione List non reversibile di blocchi funzione standard, consultare *Principi di programmazione dei blocchi funzione standard, p. 395*.

Istruzioni non equivalenti da evitare

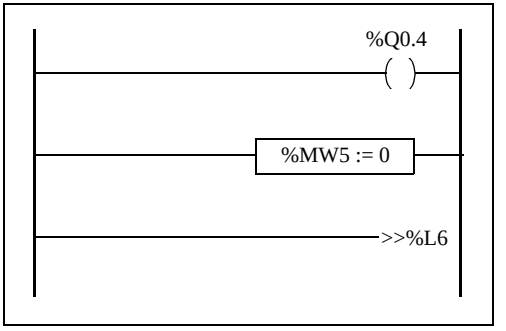
Evitare di utilizzare determinate istruzioni List o alcune combinazioni di istruzioni e operandi, che non hanno equivalenti nei linguaggi Ladder. Ad esempio, l'istruzione N (inverte il valore nell'accumulatore booleano) non presenta un'istruzione Ladder equivalente.

La tabella che segue identifica tutte le istruzioni di programmazione List non reversibili in Ladder.

Istruzione List	Operando	Descrizione
JMPCN	%Li	Salto Not condizionale
N	nessuno	Negazione (Not)
ENDCN	nessuno	Fine Not condizionale

Reti non condizionali

Per programmare reti non condizionali, è necessario seguire le raccomandazioni di programmazione List riportate di seguito, al fine di garantire la reversibilità tra linguaggio List e linguaggio Ladder. Le reti non condizionali non sono soggette ad alcun test o condizione. Le uscite o le istruzioni di azione sono sempre attivate o eseguite.
Il diagramma che segue fornisce esempi di reti non condizionali e la sequenza List equivalente.




```
LD      1
ST      %Q0.4
LD      1
[%MW5 := 0]
JMP     %L6
```

Si noti che ognuna delle sequenze List non condizionali riportate sopra inizia con un'istruzione di caricamento seguita da un 1, ad eccezione dell'istruzione JMP. Questa combinazione imposta il valore dell'accumulatore booleano a 1, imposta conseguentemente la bobina (istruzione di memorizzazione) a 1 e imposta %MW5 a 0 a ogni scansione del programma. L'eccezione è l'istruzione di salto List non condizionale (JMP %L6), che viene eseguita indipendentemente dal valore dell'accumulatore e non richiede l'impostazione a uno dell'accumulatore.

Rete Ladder/List

Se si converte un programma List non completamente reversibile, le porzioni reversibili vengono visualizzate nella vista Ladder e quelle non reversibili nelle Reti List Ladder.
Una rete List Ladder funziona esattamente come un piccolo editor List. Permette infatti all'utente di visualizzare e modificare le parti irreversibili di un programma Ladder.

Documentazione del programma

Documentazione del programma

È possibile documentare il programma inserendo commenti tramite gli editor List e Ladder.

- Utilizzare l'editor List per documentare il programma con Commenti di riga List. Questi commenti possono apparire sulla stessa riga delle istruzioni di programmazione o su altre righe.
- Utilizzare l'editor Ladder per documentare il programma con intestazioni di rete direttamente sulla rete. Queste intestazioni si trovano immediatamente sopra la rete.

Il software di programmazione TwidoSoft utilizza questi commenti ai fini della reversibilità. Durante la conversione di un programma List in Ladder, TwidoSoft utilizza alcuni commenti List per creare l'intestazione della rete. A questo scopo, i commenti inseriti tra le sequenze List sono utilizzati come intestazioni della rete.

Esempio di commenti di riga List

Di seguito è riportato un esempio di programma List con commenti di riga List.

```
---- ( * TITOLO DELL'INTESTAZIONE DELLA RETE 0 * )
---- ( * PRIMO COMMENTO DELL'INTESTAZIONE DELLA RETE 0 * )
---- ( * SECONDO COMMENTO DELL'INTESTAZIONE DELLA RETE 0 * )
     0   LD  % I0. 0  ( * QUESTA È UNA RIGA DI COMMENTO *)
     1   OR  %I0. 1  ( * RIGA DI COMMENTO IGNORATA IN CASO DI CONVERSIONE IN
    LADDER * )
     2   ANDM  %M10
     3   ST        M101
---- ( * INTESTAZIONE DELLA RETE 1 * )
---- ( * QUESTA RETE CONTIENE UN'ETICHETTA * )
---- ( * SECONDO COMMENTO DELL'INTESTAZIONE DELLA RETE 1 * )
---- ( * TERZO COMMENTO DELL'INTESTAZIONE DELLA RETE 1 * )
---- ( * QUARTO COMMENTO DELL'INTESTAZIONE DELLA RETE 1 * )
     4   % L5:
     5   LD  %M101
     6   [ %MW20 := %KW2 * 16 ]
---- ( * QUESTA RETE CONTIENE UN TITOLO DI INTESTAZIONE * )
     7   LD  %Q0. 5
     8   OR  %I0.  3
     9   ORR  I0.  13
    10   ST  %Q0.5
```

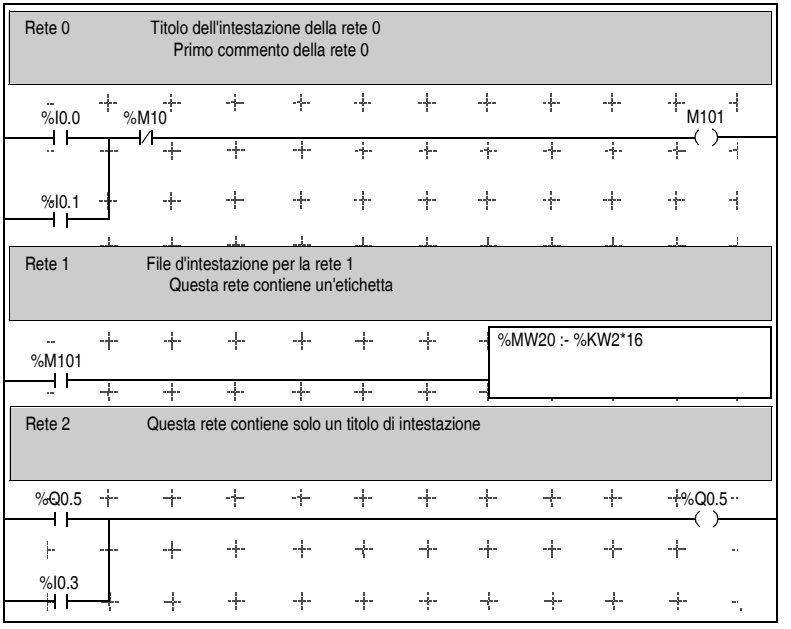
Conversione di commenti List in Ladder

Se si convertono le istruzioni List in linguaggio Ladder, i commenti di riga List vengono visualizzati nell'editor Ladder in base alle regole seguenti:

- Il primo commento da solo su una riga viene considerato come intestazione di rete.
- Eventuali altri commenti dopo il primo diventano il corpo della rete.
- Dopo aver occupato le righe del corpo dell'intestazione, i commenti di riga rimanenti tra le sequenze List vengono ignorati, come altri commenti sulle righe List contenenti istruzioni List.

Esempio di commenti di intestazione di rete

Di seguito è riportato un esempio di un programma Ladder con commenti di intestazione di rete.



Conversione di commenti Ladder in List

Se si converte il linguaggio Ladder in istruzioni List, i commenti di intestazione di rete vengono visualizzati nell'editor List in base alle regole seguenti:

- I commenti di intestazione di rete vengono inseriti tra le sequenze List associate.
- Tutte le etichette (%Li:) o le dichiarazioni di subroutine (SRi:) sono collocate sulla riga che segue l'intestazione e precedono immediatamente la sequenza List.
- Se si è convertita una sequenza List in Ladder, eventuali commenti ignorati vengono visualizzati nell'editor List.

Lista d'istruzioni, linguaggio

14

In breve

Oggetto di questo capitolo

Questo capitolo descrive la programmazione con il linguaggio List.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Panoramica dei programmi List	350
Funzionamento delle istruzioni List	352
Istruzioni linguaggio List	353
Utilizzo delle parentesi	356
Istruzioni Stack (MPS, MRD, MPP)	358

Panoramica dei programmi List

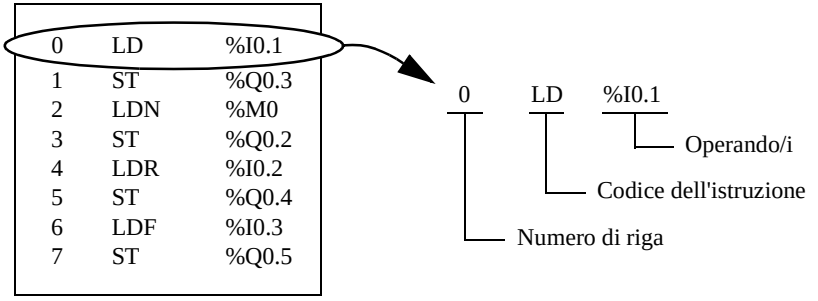
Introduzione

Un programma scritto in linguaggio List è costituito da una serie di istruzioni eseguite sequenzialmente dal controller. Ogni istruzione List è rappresentata da una singola riga di programma e include tre componenti:

- Numero di riga
- Codice dell'istruzione
- Operando/i

Esempio di un programma List

Di seguito è riportato un esempio di un programma List.



Numero di riga

I numeri di riga sono generati automaticamente quando si immette un'istruzione. Le righe vuote e le righe di commento non hanno numeri di riga.

Codice dell'istruzione

Il codice dell'istruzione è il simbolo di un operatore che identifica l'operazione da eseguire sull'operando o sugli operandi. Operatori tipici sono quelli che specificano operazioni Booleane o numeriche.

Nell'esempio di programma riportato sopra, ad esempio, LD è l'abbreviazione del codice di istruzione di un'istruzione LOAD. L'istruzione LOAD carica il valore dell'operando %I0.1 in un registro interno chiamato accumulatore.

Sono disponibili due tipi di istruzioni di base:

- Istruzioni di test
  Queste istruzioni impostano o testano le condizioni necessarie per eseguire un'azione. Ad esempio, LOAD (LD) e AND.
- Istruzioni di azione
  Queste istruzioni eseguono azioni che sono il risultato delle condizioni di impostazione. Si tratta, ad esempio, delle istruzioni di assegnazione quali STORE (ST) e RESET (R).

Operando

Un operando è un numero, indirizzo o simbolo che rappresenta un valore manipolabile da un programma tramite un'istruzione. Nell'esempio di programma precedente, l'operando %I0.1 è un indirizzo assegnato al valore di un ingresso del controller. A seconda del tipo di codice, un'istruzione può avere da zero a tre operandi.

Gli operandi possono rappresentare:

- Gli ingressi e le uscite del controller quali ad esempio sensori, pulsanti e relè.
- Funzioni di sistema predefinite come i timer e i contatori.
- Operazioni aritmetiche, logiche, di confronto e numeriche.
- Variabili interne del controller come bit e parole.

Funzionamento delle istruzioni List

Introduzione

Le istruzioni List presentano un solo operando esplicito, l'altro è implicito. L'operando implicito è il valore dell'accumulatore booleano. Ad esempio, nell'istruzione LD %I0.1, %I0.1 è l'operando esplicito. L'operando implicito è memorizzato nell'accumulatore e viene sovrascritto dal valore di %I0.1.

Funzionamento

L'istruzione List esegue un'operazione specificata sul contenuto dell'accumulatore e dell'operatore esplicito e sostituisce il contenuto dell'accumulatore con il risultato. Ad esempio, l'operazione AND %I1.2 esegue un AND logico tra il contenuto dell'accumulatore e l'Ingresso 1.2 e sostituisce il contenuto dell'accumulatore con questo risultato.
Tutte le istruzioni booleane, eccetto Load, Store e Not, operano su due operandi. Il valore dei due operandi può essere Vero o Falso e l'esecuzione di programma delle istruzioni produce un valore singolo: Vero o Falso. Le istruzioni Load inseriscono il valore dell'operando nell'accumulatore, mentre le istruzioni Store trasferiscono il valore dall'accumulatore all'operando. L'istruzione Not non presenta operandi espliciti e inverte semplicemente lo stato dell'accumulatore.

Istruzioni List supportate

La tabella seguente mostra alcuni esempi istruzioni in linguaggio List.

Tipo di istruzione	Esempio	Funzione
Istruzione Bit	LD %M10	Legge il bit interno %M10
Istruzione blocco	IN %TM0	Avvia il timer %TM0
Istruzione parola	[%MW10 := %MW50+100]	Operazione di addizione
Istruzione di programma	SR5	Richiama la subroutine n. 5
Istruzione Grafcet	-*-8	Passo 8

Istruzioni linguaggio List

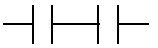
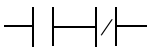
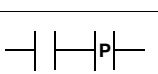
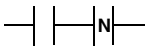
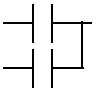
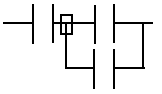
Introduzione

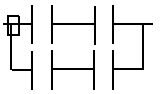
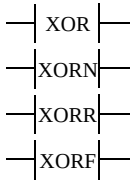
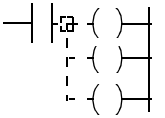
Il linguaggio List comprende i seguenti tipi di istruzioni:
- Istruzioni di test
- Istruzioni di azione
- Istruzioni di blocco funzione

Questa sezione identifica e descrive le istruzioni Twido per la programmazione List.

Istruzioni di test

La tabella che segue descrive le istruzioni di test nel linguaggio List.

Nome	Elemento grafico equivalente	Funzione
LD		Il risultato booleano è uguale allo stato dell'operando.
LDN		Il risultato booleano è uguale all'inverso dello stato dell'operando.
LDR		Il risultato booleano passa a 1 dopo il rilevamento dell'operando (fronte di salita) passando da 0 a 1.
LDF		Il risultato booleano passa a 0 dopo il rilevamento dell'operando (fronte di discesa) passando da 1 a 0.
AND		Il risultato booleano è uguale all'AND logico tra il risultato booleano dell'istruzione precedente e lo stato dell'operando.
ANDN		Il risultato booleano è uguale all'AND logico tra il risultato booleano dell'istruzione precedente e l'inverso dello stato dell'operando.
ANDR		Il risultato booleano è uguale all'AND logico tra il risultato booleano dell'istruzione precedente e il rilevamento del fronte di salita dell'operando (1 = fronte di salita).
ANDF		Il risultato booleano è uguale all'AND logico tra il risultato booleano dell'istruzione precedente e il rilevamento del fronte di discesa dell'operando (1 = fronte di discesa).
OR		Il risultato booleano è uguale all'OR logico tra il risultato booleano dell'istruzione precedente e lo stato dell'operando.
AND(		AND logico (8 livelli di parentesi)

Nome	Elemento grafico equivalente	Funzione
OR(		OR logico (8 livelli di parentesi)
XOR, XORN, XORR, XORF		OR esclusivo
MPS MRD MPP		Commutazione alle bobine.
N	-	Negazione (NOT)

Istruzioni di azione

La tabella che segue descrive le istruzioni di azione nel linguaggio List.

Nome	Elemento grafico equivalente	Funzione
ST	—( )—	L'operando associato assume il valore del risultato dell'area di test.
STN	—(/)—	L'operando associato assume il valore inverso del risultato dell'area di test.
S	—(S)—	L'operando associato viene impostato a 1 quando il risultato dell'area di test è 1.
R	—(R)—	L'operando associato viene impostato a 0 quando il risultato dell'area di test è 1.
JMP	->>%Li	Collega non condizionalmente a una sequenza definita da etichetta, precedente o successiva nel programma.
SRn	->>%SRi	Connessione all'inizio di una subroutine.
RET	<RET>	Ritorno da una subroutine.
END	<END>	Fine del programma.
ENDC	<ENDC>	Fine del programma condizionato con il risultato booleano di 1.
ENDCN	<ENDCN>	Fine del programma condizionato con il risultato booleano di 0.

Istruzioni di blocco funzione

La tabella che segue descrive i blocchi funzione nel linguaggio List.

Nome	Elemento grafico equivalente	Funzione
Timer, contatori, registri e così via.		Per ciascun blocco funzione, esistono istruzioni per il controllo del blocco. Viene utilizzato un formato strutturato per collegare direttamente gli ingressi e le uscite del blocco. **Nota:** Le uscite dei blocchi funzione non possono essere collegate tra loro (corti verticali).

Utilizzo delle parentesi

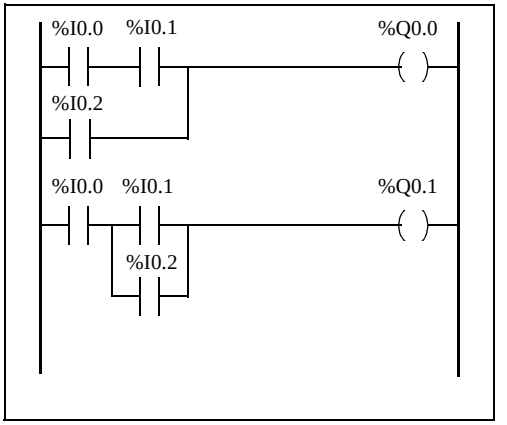
Introduzione

Nelle istruzioni logiche AND e OR, le parentesi permettono di specificare divergenze nei programmi Ladder. Le parentesi sono associate a istruzioni e vengono utilizzate come segue:

- L'apertura della parentesi è associata all'istruzione AND oppure OR.
- La chiusura della parentesi corrisponde a un'istruzione richiesta per ogni parentesi aperta.

Esempio di utilizzo dell'istruzione AND

I diagrammi che seguono mostrano alcuni esempi di impiego delle parentesi in un'istruzione AND. AND(...).

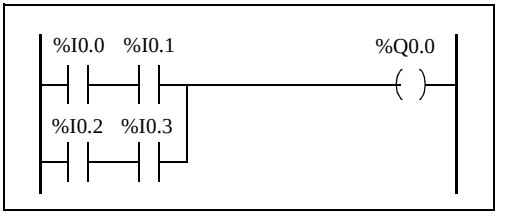



```
LD      %I0.0
AND     %I0.1
OR      %I0.2
ST      %Q0.0

LD      %I0.0
AND(    %I0.1
OR      %I0.2
)
ST      %Q0.1
```

Esempio di utilizzo dell'istruzione OR

I diagrammi che seguono mostrano alcuni esempi di impiego delle parentesi in un'istruzione OR. OR(...).




```
LD      %I0.0
AND     %I0.1
OR(     %I0.2
AND     %I0.3
)
ST      %Q0.0
```

Modificatori

La tabella che segue elenca i modificatori che è possibile assegnare alle parentesi.

Modificatore	Funzione	Esempio
N	Negazione	AND(N o OR(N
F	Fronte di discesa	AND(F o OR(F
R	Fronte di salita	AND(R o OR(R
[	Confronto	Vedere *Istruzioni di confronto, p. 422*

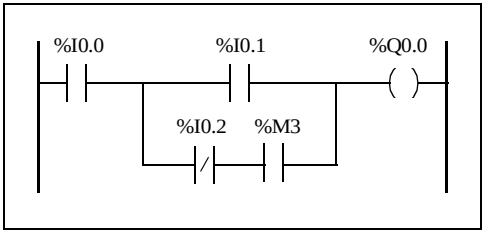
Annidamento delle parentesi

È possibile annidare al massimo otto livelli di parentesi.
Per annidare le parentesi, osservare le seguenti regole:

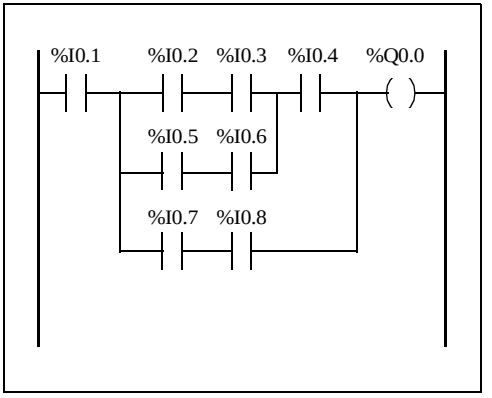
- Ogni parentesi aperta deve avere una corrispondente parentesi chiusa.
- Etichette (%Li:), subroutine (SRi:), istruzioni di salto(JMP) e istruzioni di blocco funzione non devono essere inserite in espressioni tra parentesi.
- Le istruzioni Store ST, STN, S e R non devono essere programmate tra parentesi.
- Le istruzioni di stack MPS, MRD e MPP non possono essere utilizzate tra parentesi.

Esempi di parentesi annidate

I diagrammi che seguono mostrano esempi di parentesi annidate.




```
LD      %I0.0
AND(    %I0.1
OR(N    %I0.2
AND     %M3
)
)
ST      %Q0.0
```




```
LD      %I0.1
AND(    %I0.2
AND     %I0.3
OR(     %I0.5
AND     %I0.6
)
AND     %I0.4
OR(     %I0.7
AND     %I0.8
)
)
ST      %Q0.0
```

Istruzioni Stack (MPS, MRD, MPP)

Introduzione

Le istruzioni Stack elaborano l'instradamento alle bobine. Le istruzioni MPS, MRD e MPP utilizzano un'area di memorizzazione temporanea denominata stack che può contenere fino a otto espressioni booleane.

Nota: Tali istruzioni non possono essere utilizzate all'interno di un'espressione tra parentesi.

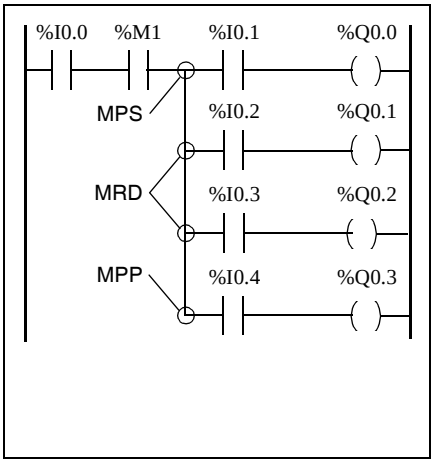
Funzionamento delle istruzioni Stack

La tabella che segue descrive il funzionamento delle tre istruzioni Stack.

Istruzione	Descrizione	Funzione
MPS	Memory Push onto stack	Memorizza il risultato dell'ultima istruzione logica (contenuto dell'accumulatore) in alto nello stack e sposta gli altri valori al fondo dello stack.
MRD	Memory Read from stack	Legge il contenuto nella parte superiore dello stack nell'accumulatore.
MPP	Memory Pop from stack	Copia il valore della parte superiore dello stack nell'accumulatore e sposta gli altri valori verso l'alto.

Esempi di istruzioni Stack

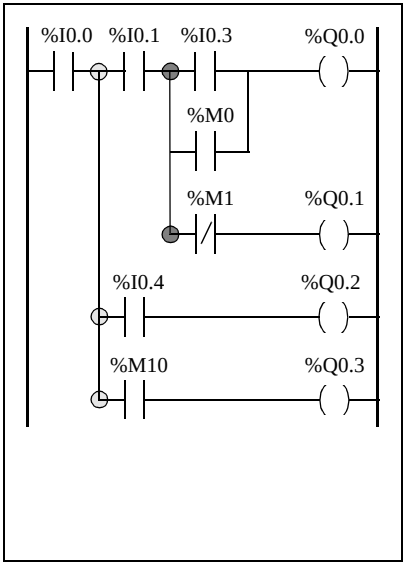
I diagrammi che seguono mostrano alcuni esempi di impiego delle istruzioni Stack.



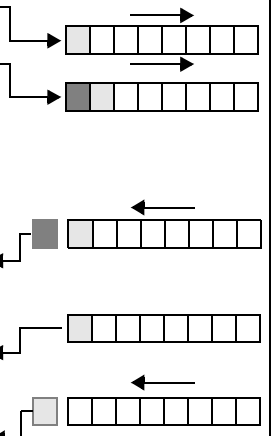

```
LD      %I0.0
AND     %M1
MPS
AND     %I0.1
ST      %Q0.0
MRD
AND     %I0.2
ST      %Q0.1
MRD
AND     %I0.3
ST      %Q0.2
MPP
AND     %I0.4
ST      %Q0.3
```

Esempi di funzionamento Stack

I diagrammi che seguono mostrano il funzionamento delle istruzioni stack.




```
LD      %I0.0
MPS
AND     %I0.1
MPS
AND(    %I0.3
OR      %M0
)
ST      %Q0.0
MPP
ANDN    %M1
ST      %Q0.1
MRD
AND     %I0.4
ST      %Q0.2
MPP
AND     %M10
ST      %Q0.3
```



Grafcet

15

In breve

Oggetto di questo capitolo

Questo capitolo descrive la programmazione con il linguaggio Grafcet.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Descrizione delle istruzioni Grafcet	362
Descrizione della struttura di programma Grafcet	366
Azioni associate ai passi Grafcet	369

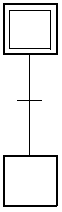
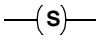
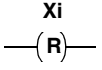
Descrizione delle istruzioni Grafcet

Introduzione

Le istruzioni Grafcet in TwidoSoft costituiscono un metodo semplice per la conversione di una sequenza di controllo (grafico Grafcet).
Il numero massimo di passi Grafcet dipende dal tipo di controller Twido. Il numero di passi attivi contemporaneamente è limitato solo dal numero di passi totale.
Per TWDLCAA10DRF e TWDLCAA16DRF, sono disponibili i passi da 1 a 62. I passi 0 e 63 sono riservati per la pre- e post-elaborazione. Per tutti gli altri controller, sono disponibili i passi da 1 a 95.

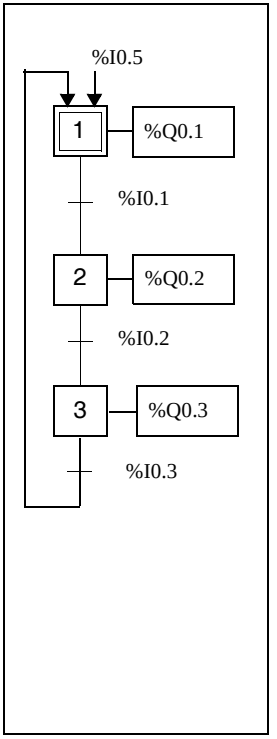
Istruzioni Grafcet

La tabella che segue elenca tutte le istruzioni e gli oggetti richiesti per programmare un grafico Grafcet.

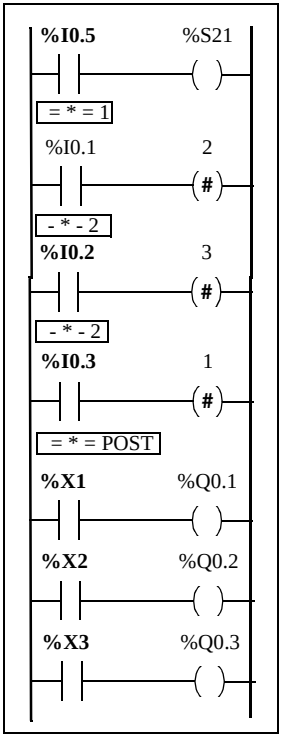
Rappresentazione grafica (1)	Trascrizione in linguaggio TwidoSoft	Funzione
Illustrazione:  **Passo iniziale** **Transizione** **Fase**	=*= i	Avvia il passo iniziale (2)
	# i	Attiva il passo i dopo aver disabilitato il passo corrente
	-*- i	Avvia il passo i e convalida la transizione associata (2)
	#	Disabilita il passo corrente senza attivare altri passi
	#Di	Disabilita il passo i e il passo corrente
	=*= POST	Avvia la post-elaborazione e termina l'elaborazione sequenziale
	%Xi	Il bit associato al passo i può essere provato e scritto (il numero massimo di passi dipende dal controller)
Xi 	LD %Xi, LDN %Xi AND %Xi, ANDN %Xi, OR %Xi, ORN %Xi XOR %Xi, XORN %Xi	Prova l'attività del passo i
Xi  **S**	S %Xi	Attiva il passo i
Xi  **R**	R %Xi	Disabilita il passo i

(1) La rappresentazione grafica non è supportata.
(2) Il primo passo =*=i o -*-i scritto indica l'avvio dell'elaborazione sequenziale e quindi la fine della pre-elaborazione.

Esempi Grafcet Sequenza lineare:



Non supportato

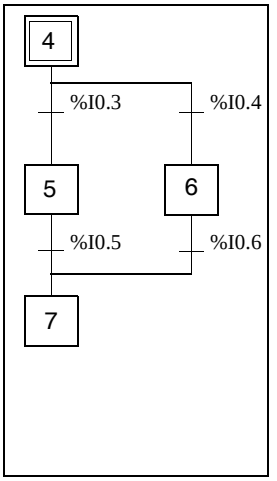


Programma
Ladder Twido

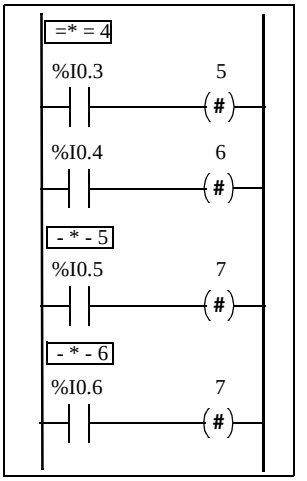
```
LD      %I0.5
ST      %S21
=*=     1
LD      %I0.1
 #      2
-*-     2
LD      %I0.2
 #      3
-*-     3
LD      %I0.3
 #      1
=*=     POST
LD      %X1
ST      %Q0.1
LD      %X2
ST      %Q0.2
LD      %X3
ST      %Q0.3
```

Programma
List Twido

Sequenza alternativa:



Non supportato



Programma
Ladder Twido

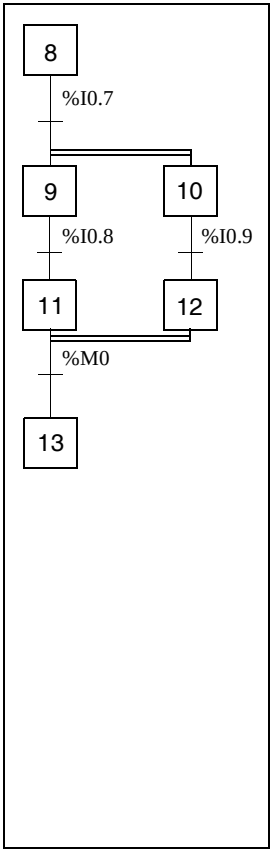
```
=*=     4
 LD     %I0.3
 #      5
 LD     %I0.4
 #      6

 -*-    5
 LD     %I0.5
 #      7

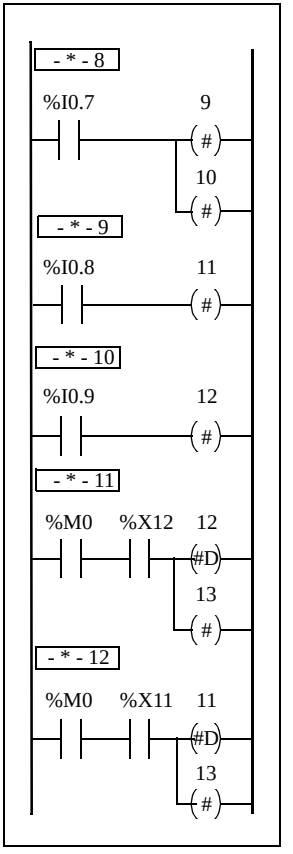
-*-     6
 LD     %I0.6
 #      7
```

Programma
List Twido

Sequenze simultanee:



Non supportato



Programma
Ladder Twido

```
 -*-      8
LD        %I0.7
#         9
#         10

-*-       9
LD        %I0.8
#         11

-*-       10
LD        %I0.9
#         12

-*-       11
LD        %M0
AND       %X12
#D        12
#         13

-*-       12
LD        %M0
AND       %X11
#D        11
#         13
```

Programma
List Twido

Nota: per rendere operativo un grafico Grafcet, occorre dichiarare almeno un passo attivo tramite l'istruzione =*=i (passo iniziale) oppure si deve preposizionare il grafico durante la pre-elaborazione utilizzando il bit di sistema %S23 e l'istruzione S %Xi.

Descrizione della struttura di programma Grafcet

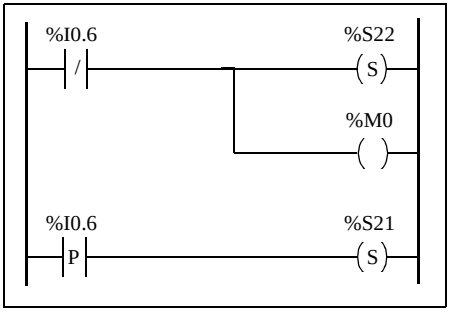
Introduzione

Il programma Grafcet TwidoSoft è composto da tre parti:
- Pre-elaborazione
- Elaborazione sequenziale
- Post elaborazione

Pre-elaborazione

La pre-elaborazione comprende:
- Ritorni di alimentazione
- Errori
- Cambiamenti di modalità operativa
- Preposizionamento dei passi Grafcet
- Logica di ingresso

Il fronte di salita dell'ingresso %I0.6 imposta a 1 il bit %S21. In tal modo si disattivano i passi attivi e si attivano i passi iniziali.




```
000     LDN     %I0.6
001     S       %S22
002     ST      %M0
003     LDR     %I0.6
004     S       %S21
```

La pre-elaborazione inizia dalla prima linea del programma e termina alla prima occorrenza di un'istruzione "= * =" o "- * -".
Al controllo Grafcet sono dedicati tre bit di sistema. %S21, %S22 e %S23. Questi bit di sistema sono impostati a 1 (se necessario) dall'applicazione, in genere nella pre-elaborazione. La funzione associata viene eseguita dal sistema al termine della pre-elaborazione e il bit di sistema viene quindi azzerato dal sistema.

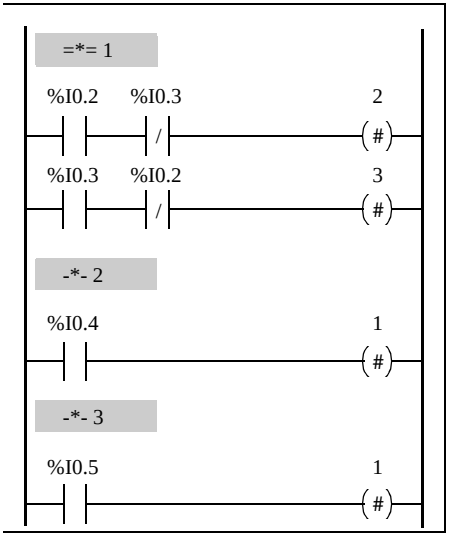
Bit di sistema	Nome	Descrizione
%S21	Inizializzazione Grafcet	Tutti i passi attivi vengono disattivati e attivati i passi iniziali.
%S22	Reinizializzazione Grafcet	Tutti i passi vengono disattivati.
%S23	Preposizionamento Grafcet	Impostare questo bit a 1 se %Xi sono scritti esplicitamente dall'applicazione in pre-elaborazione. Se questo bit viene mantenuto a 1 dalla pre-elaborazione senza alcuna modifica esplicita degli oggetti %Xi, Grafcet viene bloccato (non viene considerato alcun aggiornamento).

Elaborazione sequenziale

L'elaborazione sequenziale avviene nel grafico (istruzioni che rappresentano il grafico).

- Passi
- Azioni associate ai passi
- Transizioni
- Condizioni di transizione

Esempio:




```
005     =*=     1
006     LD      %I0.2
007     ANDN    %I0.3
008     #       2
009     LD      %I0.3
010     ANDN    %I0.2
011     #       3
012     -*-     2
013     LD      %I0.4
014     #       1
015     -*-     3
016     LD      %I0.5
017     #       1
```

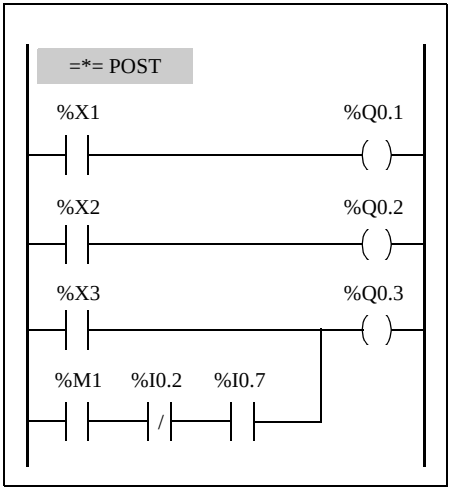
L'elaborazione sequenziale termina con l'esecuzione dell'istruzione "= * = POST" o al termine del programma.

Post elaborazione

La post elaborazione comprende:

- Comandi dall'elaborazione sequenziale per il controllo delle uscite
- Blocchi di sicurezza specifici delle uscite

Esempio:




```
018     =*=     POST
019     LD      %X1
020     ST      %Q0.1
021     LD      %X2
022     ST      %Q0.2
023     LD      %X3
024     OR(     %M1
025     ANDN    %I0.2
026     AND     %I0.7
027     )
028     ST      %Q0.3
```

Azioni associate ai passi Grafcet

Introduzione

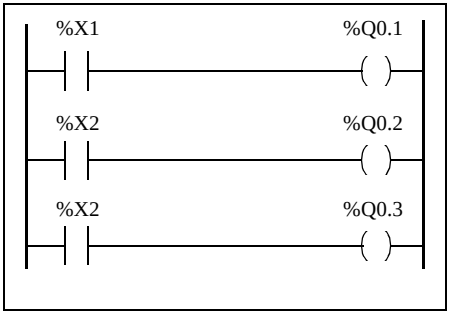
Il programma Grafcet TwidoSoft offre due metodi per programmare le azioni associate ai passi:

- Nella sezione di post elaborazione
- Nelle istruzioni List o reti Ladder dei passi stessi

Associazione delle azioni nella post elaborazione

Se sono presenti vincoli della modalità di esecuzione o sicurezza, è opportuno programmare le azioni nella sezione di post elaborazione di un'applicazione Grafcet. È possibile utilizzare le istruzioni List Set e Reset o alimentare le bobine in un programma Ladder per attivare i passi Grafcet (%Xi).

Esempio:

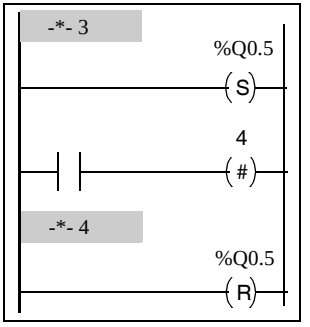



```
018    =*=    POST
019    LD     %X1
020    ST     %Q0.1
021    LD     %X2
022    ST     %Q0.2
023    LD     %X3
024    ST     %Q0.3
```

Associazione di azioni da un'applicazione

È possibile programmare le azioni associate ai passi nelle istruzioni List o reti Ladder. In questo caso, l'istruzione List o la rete Ladder non viene sottoposta a scansione se il passo non è attivo. Questo è il metodo più efficace, leggibile e mantenibile per utilizzare Grafcet.

Esempio:




```
020   -*-   3
021   LD    1
022   S     %Q0.5
023   LD    %M10
024   #     4
025   -*-   4
026   LD    1
027   R     %Q0.5
028   ...
029   ...
```

Descrizione di istruzioni e funzioni

IV

In breve

Contenuto della sezione

Questa sezione descrive in modo dettagliato le istruzioni di base e avanzate, i bit e le parole di sistema dei linguaggi Twido.

Contenuto di questa parte

Questa parte contiene i seguenti capitoli:

Capitolo	Titolo del capitolo	Pagina
16	Istruzioni di base	373
17	Istruzioni avanzate	441
18	Bit di sistema e parole di sistema	603

Istruzioni di base

16

In breve

Oggetto di questo capitolo

Questo capitolo descrive in modo dettagliato le istruzioni e i blocchi funzione utilizzati per creare programmi di regolazione elementari dei controller Twido.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
16.1	Elaborazione booleana	375
16.2	Blocchi funzione di base	392
16.3	Elaborazione numerica	415
16.4	Istruzioni di programma	434

16.1 Elaborazione booleana

In breve

Oggetto di questa sezione

Questa sezione presenta un'introduzione all'elaborazione booleana. Fornisce inoltre descrizioni e indicazioni di programmazione per le istruzioni booleane.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Istruzioni booleane	376
Informazioni sul formato per la descrizione delle istruzioni booleane	378
Istruzioni Load (LD, LDN, LDR, LDF)	380
Istruzioni d'assegnazione (ST, STN, R, S)	382
Istruzioni AND logico (AND, ANDN, ANDR, ANDF)	384
Istruzioni OR logico (OR, ORN, ORR, ORF)	386
OR esclusivo, istruzioni (XOR, XORN, XORR, XORF)	388
Istruzione NOT (N)	390

Istruzioni booleane

Introduzione

Le istruzioni booleane possono essere confrontate con gli elementi grafici del linguaggio Ladder, come elencato nella tabella seguente.

Elemento	Istruzione	Esempio	Descrizione
Elementi di test	L'istruzione Load (LD) è equivalente a un contatto aperto.	LD %I0.0	Il contatto è chiuso quando il bit %I0.0 è allo stato 1.
Elementi d'azione	L'istruzione Store (ST) è equivalente a una bobina.	ST %Q0.0	L'oggetto bit associato assume il valore logico dell'accumulatore del bit (risultato della logica precedente)

Il risultato booleano degli elementi di test viene applicato agli elementi di azione come indicato dalle istruzioni seguenti.

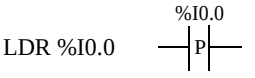
```
LD    %I0.0
AND   %I0.1
ST    %Q0.0
```

Prova ingressi del controller

È possibile utilizzare le istruzioni di test booleane per individuare i fronti di salita o discesa agli ingressi del controller. Il fronte viene individuato quando lo stato di un ingresso passa dal valore "scansione n-1" al valore "scansione n" corrente e tale rilevamento permane durante la scansione corrente.

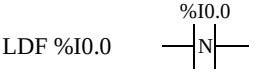
Fronte di salita, rilevamento

L'istruzione LDR (Load Rising Edge) è equivalente a un contatto di rilevamento del fronte di salita. Il fronte di salita individua il passaggio del valore dell'ingresso da 0 a 1.
Un contatto di rilevazione di transizione positivo viene utilizzato per individuare il fronte di salita come indicato nel diagramma seguente.



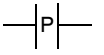
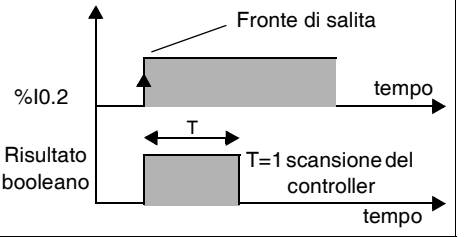
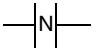
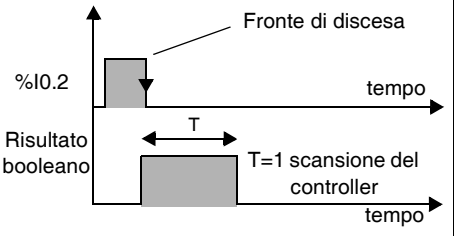
Rilevamento di un fronte di discesa

L'istruzione LDF (Load Falling Edge) è equivalente a un contatto di rilevamento del fronte di discesa. Il fronte di discesa individua il passaggio del valore dell'ingresso di controllo da 1 a 0.
Un contatto di rilevazione di transizione negativo viene utilizzato per individuare il fronte di discesa come indicato nel diagramma seguente.



Rilevamento di un fronte

La tabella che segue riassume le istruzioni di rilevamento dei fronti e i relativi tempi:

Fronte	Istruzione di test	Ladder	Diagramma dei tempi
Fronte di salita	LDR %I0.0		
Fronte di discesa	LDF %I0.0		

Nota: è possibile applicare le istruzioni sul fronte ai bit interni %Mi.

Informazioni sul formato per la descrizione delle istruzioni booleane

Introduzione

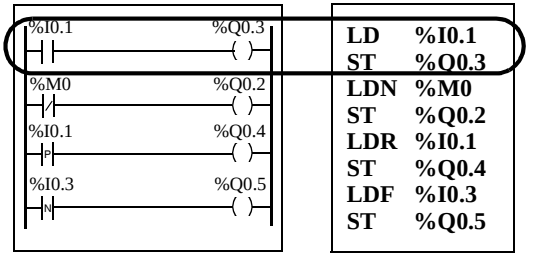
Ogni istruzione booleana in questa sezione viene descritta con le informazioni seguenti:

- Breve descrizione
- Esempio dell'istruzione e linguaggio Ladder corrispondente
- Elenco degli operandi permessi
- Diagramma dei tempi

Le spiegazioni che seguono forniscono ulteriori dettagli sulla descrizione delle istruzioni booleane in questa sezione.

Esempi

L'illustrazione che segue mostra come vengono forniti gli esempi per ogni istruzione.



Equivalenti linguaggio Ladder | Istruzioni List

Operandi permessi

La tabella che segue definisce i tipi di operandi permessi utilizzati per le istruzioni booleane.

Operando	Descrizione
0/1	Valore immediato di 0 o 1
%I	Ingresso controller %Ii.j
%Q	Uscita controller %Qi.j
%M	Bit interno %Mi
%S	Bit di sistema %Si
%X	Bit di passo %Xi
%BLK.x	Bit blocco funzione (ad esempio, %TMi.Q)
%•:Xk	Bit parola (ad esempio, %MWi:Xk)
[	Espressione di confronto (ad esempio, [%MWi<1000])

Diagrammi dei tempi

L'illustrazione che segue mostra come vengono visualizzati i diagrammi dei tempi per ogni istruzione.

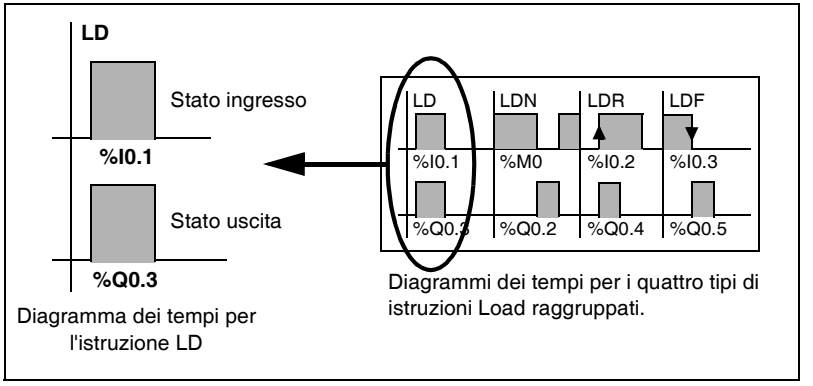


Diagramma dei tempi per l'istruzione LD

Diagrammi dei tempi per i quattro tipi di istruzioni Load raggruppati.

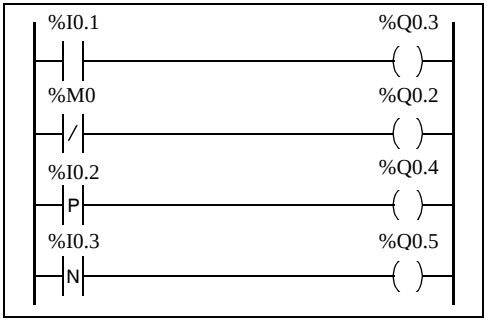
Istruzioni Load (LD, LDN, LDR, LDF)

Introduzione

Le istruzioni Load LD, LDN, LDR e LDF corrispondono rispettivamente ai contatti aperto, chiuso, fronte di salita e fronte di discesa (le istruzioni LDR e LDF sono utilizzate solo con gli ingressi e le parole interne del controller e con gli ingressi slave AS-Interface CANopen PDO).

Esempi

Gli schemi che seguono mostrano alcuni esempi delle istruzioni Load.




```
LD    %I0.1
ST    %Q0.3
LDN   %M0
ST    %Q0.2
LDR   %I0.2
ST    %Q0.4
LDF   %I0.3
ST    %Q0.5
```

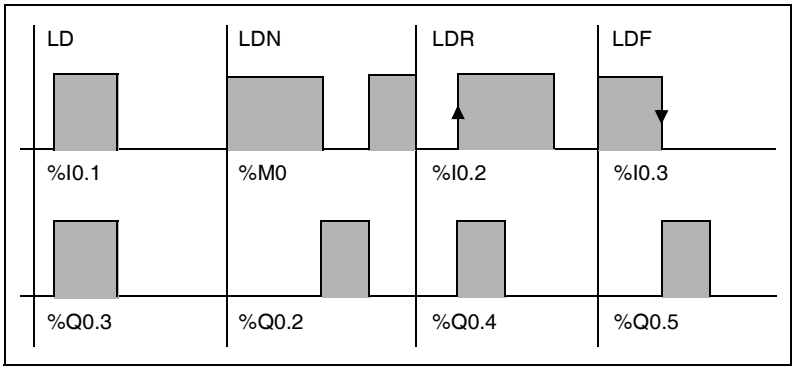
Operandi consentiti

La tabella che segue elenca i tipi di istruzioni Load con equivalenti Ladder e operandi consentiti.

Istruzione List	Equivalente Ladder	Operandi consentiti
LD		0/1, %I, %IA, %IWCx.y.z:Xk, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk,[
LDN		0/1, %I, %IA, %IWCx.y.z:Xk, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk,[
LDR		%I, %IA, %M
LDF		%I, %IA, %M

Diagramma dei tempi

L'illustrazione che segue mostra un cronogramma delle istruzioni LOAD.



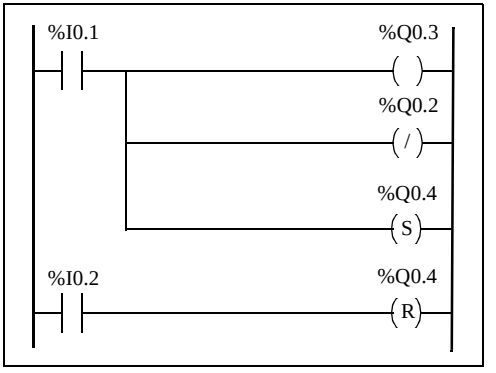
Istruzioni d'assegnazione (ST, STN, R, S)

Introduzione

Le istruzioni d'assegnazione ST, STN, S e R corrispondono rispettivamente alle bobine diretta, inversa, imposta e reimposta.

Esempi

I diagrammi che seguono mostrano alcuni esempi delle istruzioni d'assegnazione.




```
LD      %I0.1
ST      %Q0.3

STN     %Q0.2
S       %Q0.4

LD      %I0.2
R       %Q0.4
```

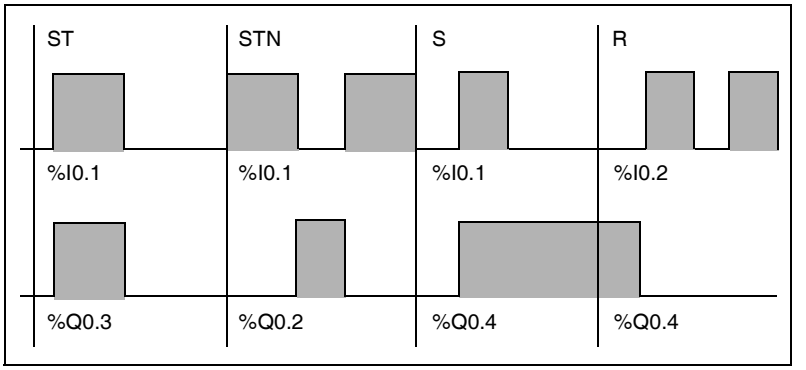
Operandi consentiti

La tabella che segue elenca i tipi di istruzioni d'assegnazione con equivalenti Ladder e operandi consentiti.

Istruzione List	Equivalente Ladder	Operandi consentiti
ST	( )	%Q,%QA,%M,%S,%BLK.x,%•:Xk
STN	(/)	%Q,%QA%M,%S,%BLK.x,%•:Xk
S	(S)	%Q,%QA,%M,%S,%X,%BLK.x,%•:Xk
R	(R)	%Q,%QA,%M,%S,%X,%BLK.x,%•:Xk

Diagramma dei tempi

Il diagramma che segue mostra i tempi delle istruzioni d'assegnazione.



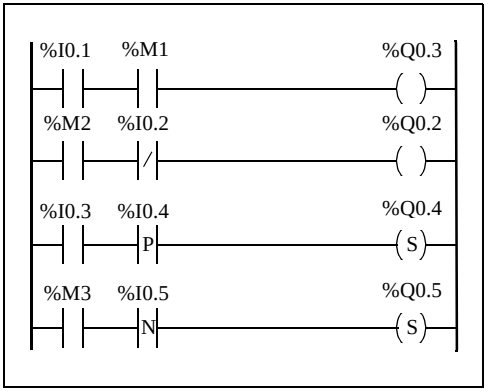
Istruzioni AND logico (AND, ANDN, ANDR, ANDF)

Introduzione

Le istruzioni AND eseguono un'operazione AND logico tra l'operando (o il suo inverso o il rispettivo fronte di salita o discesa) e il risultato booleano dell'istruzione precedente.

Esempi

I diagrammi che seguono mostrano alcuni esempi di istruzioni AND.




```
LD      %I0.1
AND     %M1
ST      %Q0.3
LD      %M2
ANDN    %I0.2
ST      %Q0.2
LD      %I0.3
ANDR    %I0.4
S       %Q0.4
LD      %M3
ANDF    %I0.5
S       %Q0.5
```

Operandi consentiti

La tabella che segue elenca i tipi di istruzioni AND con equivalenti Ladder e operandi consentiti.

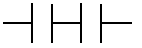
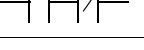
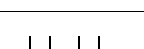
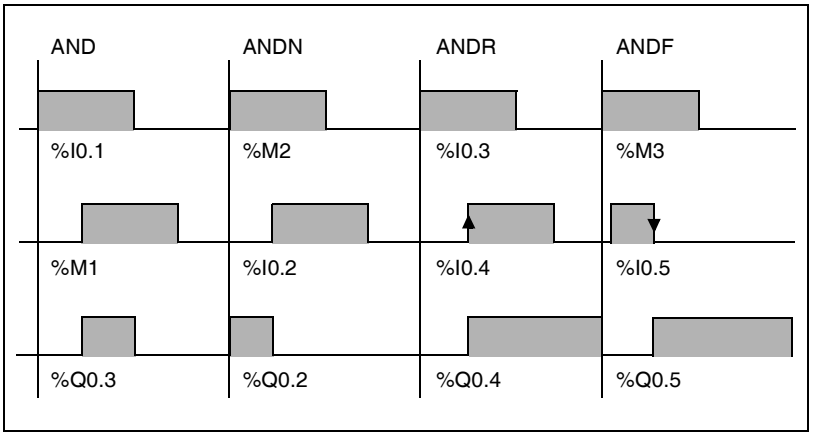
Istruzione List	Equivalente Ladder	Operandi consentiti
AND		0/1, %I, %IA, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk, [
ANDN		0/1, %I, %IA, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk, [
ANDR		%I, %IA, %M
ANDF		%I, %IA, %M

Diagramma dei tempi

Il diagramma che segue mostra i tempi delle istruzioni AND.



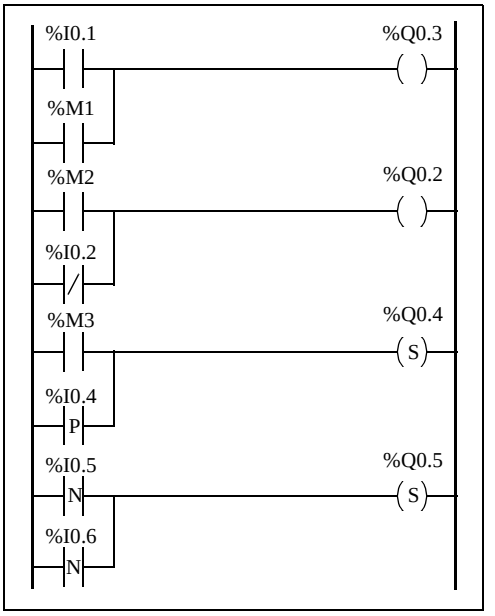
Istruzioni OR logico (OR, ORN, ORR, ORF)

Introduzione

Le istruzioni OR eseguono un'operazione OR logico tra l'operando (o il suo inverso o il rispettivo fronte di salita o discesa) e il risultato booleano dell'istruzione precedente.

Esempi

I diagrammi che seguono mostrano alcuni esempi di istruzioni OR.




```
LD      %I0.1
OR      %M1
ST      %Q0.3

LD      %M2
ORN     %I0.2
ST      %Q0.2

LD      %M3
ORR     %I0.4
S       %Q0.4

LDF     %I0.5
ORF     %I0.6
S       %Q0.5
```

Operandi consentiti

La tabella che segue elenca i tipi di istruzioni OR con equivalenti Ladder e operandi consentiti.

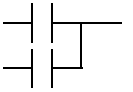
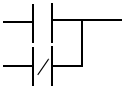
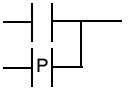
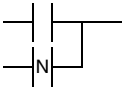
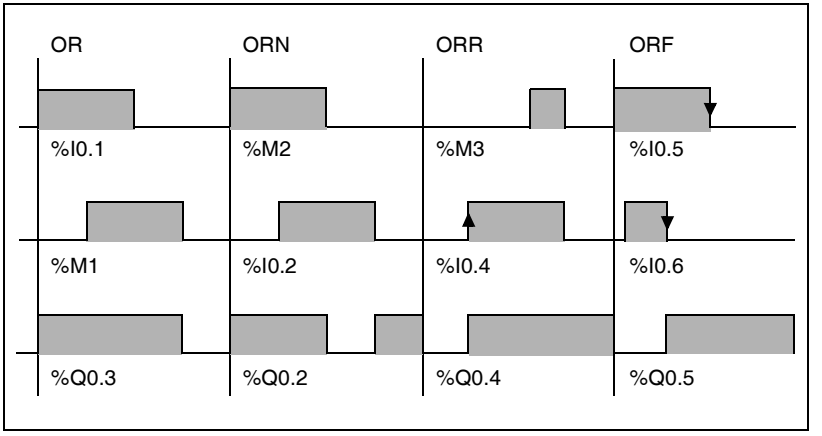
Istruzione List	Equivalente Ladder	Operandi consentiti
OR		0/1, %I,%IA, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk
ORN		0/1, %I,%IA, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk
ORR		%I, %IA, %M
ORF		%I, %IA, %M

Diagramma dei tempi

Il diagramma che segue mostra i tempi delle istruzioni OR.



OR esclusivo, istruzioni (XOR, XORN, XORR, XORF)

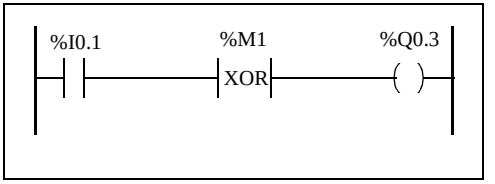
Introduzione

Le istruzioni XOR eseguono un'operazione di collegamento OR esclusivo tra l'operando (o il suo inverso o il rispettivo fronte di salita o discesa) e il risultato booleano dell'istruzione precedente.

Esempi

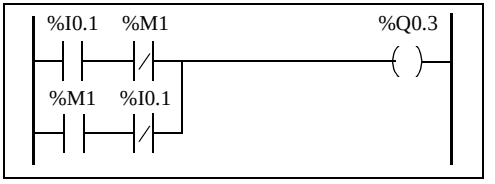
Le istruzioni XOR possono essere utilizzate come mostrato nell'esempio che segue.

Costruzione con istruzioni XOR :




```
LD      %I0.1
XOR     %M1
ST      %Q0.3
```

Costruzione senza istruzioni XOR :




```
LD      %I0.1
ANDN    %M1
OR(     %M1
ANDN    %I0.1
)
ST      %Q0.3
```

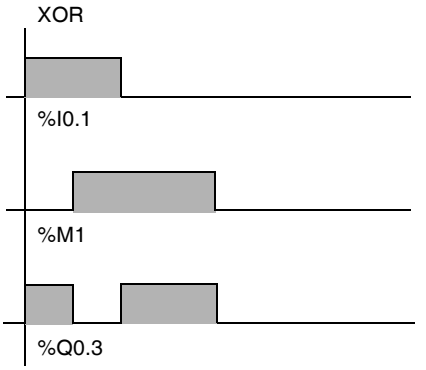
Operandi consentiti

La tabella che segue elenca i tipi di istruzioni XOR e gli operandi consentiti.

Istruzione List	Operandi consentiti
XOR	%I, %IA, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk
XORN	%I, %IA, %Q, %QA, %M, %S, %X, %BLK.x, %•:Xk
XORR	%I, %IA, %M
XORF	%I, %IA, %M

Diagramma dei tempi

Il diagramma che segue mostra i tempi delle istruzioni XOR.

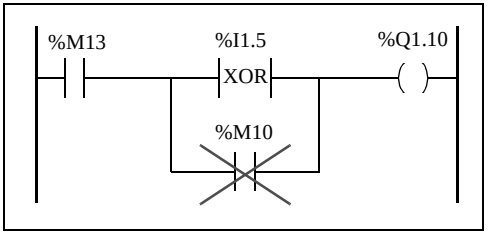


Casi speciali

Di seguito vengono indicate precauzioni speciali relative all'utilizzo delle istruzioni XOR nei programmi Ladder.

- Non inserire contatti XOR nella prima posizione di una rete.
- Non inserire contatti XOR in parallelo con altri elementi Ladder (vedere l'esempio seguente).

Come mostrato nell'esempio seguente, l'inserimento di un elemento in parallelo al contatto XOR genera un errore di convalida.



Istruzione NOT (N)

Introduzione L'istruzione NOT (N) nega il risultato booleano dell'istruzione precedente.

Esempio Di seguito è riportato un esempio dell'utilizzo dell'istruzione NOT.

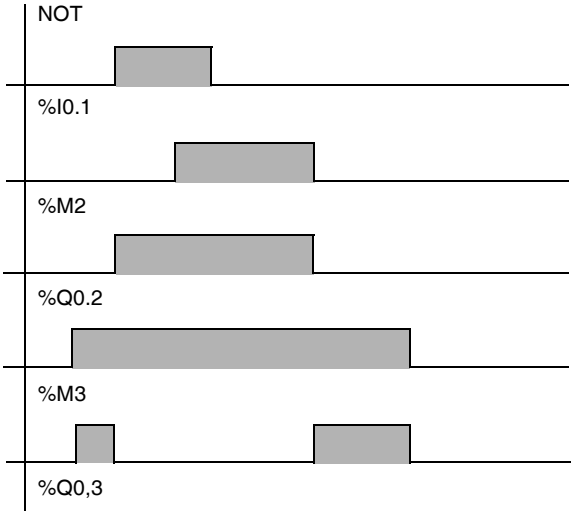
```
LD      %I0.1
OR      %M2
ST      %Q0.2
N
AND     %M3
ST      %Q0.3
```

Nota: L'istruzione NOT non è reversibile.

Operandi permessi Non applicabile.

Diagramma dei tempi

Il diagramma che segue mostra i tempi dell'istruzione NOT.



16.2 Blocchi funzione di base

In breve

Oggetto di questa sezione

Questa sezione presenta descrizioni e indicazioni di programmazione per l'utilizzo dei blocchi funzione di base.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Blocchi funzione di base	393
Principi di programmazione dei blocchi funzione standard	395
Blocco funzione timer (%TMi)	397
Timer di tipo TOF	399
Timer di tipo TON	400
Tipo di timer TP	401
Programmazione e configurazione timer	402
Blocco funzione contatore avanti/indietro (%Ci)	405
Programmazione e configurazione contatori	408
Blocco funzione registro bit di scorrimento (%SBRi)	410
Blocco funzione contatore passo passo (%SCi)	412

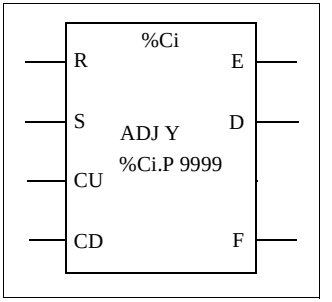
Blocchi funzione di base

Introduzione

I blocchi funzione costituiscono le origini di oggetti bit e parole specifici utilizzati dai programmi. I blocchi funzione standard presentano funzioni semplici quali timer e contatori avanti/indietro.

Esempio di un blocco funzione

Di seguito viene illustrato un esempio di blocco funzione Contatore avanti/indietro.



Blocco contatore avanti/indietro

Oggetti bit

Gli oggetti bit corrispondono alle uscite del blocco: è possibile accedere a questi bit tramite le istruzioni di test booleane utilizzando uno dei metodi seguenti:

- Direttamente (ad esempio, LD E) se sono collegati al blocco nella programmazione reversibile (vedere *Principi di programmazione dei blocchi funzione standard, p. 395*).
- Specificando il tipo di blocco (ad esempio, LD %Ci.E).

È possibile accedere agli ingressi sotto forma di istruzioni.

Oggetti parola

Gli oggetti parola corrispondono a:

- **Parametri di configurazione del blocco:**alcuni parametri sono accessibili da parte del programma (ad esempio, parametri di preselezione) e altri non lo sono (ad esempio, base tempo).
- **Valori correnti:**ad esempio, %Ci.V, il valore del conteggio corrente.

Oggetti parola e bit accessibili

La tabella che segue descrive oggetti parola e bit dei blocchi funzione di base a cui può accedere il programma.

Blocco funzione di base	Simbolo	Intervallo (i)	Tipi di oggetti	Descrizione	Indirizzo	Accesso in scrittura
Timer	%TMi	0 - 127	Parola	Valore corrente	%TMi.V	no
				Valore preimpostato	%TMi.P	sì
			Bit	Uscita timer	%TMi.Q	no
Contatore avanti/indietro	%Ci	0 - 127	Parola	Valore corrente	%Ci.V	no
				Valore preimpostato	%Ci.P	sì
			Bit	Uscita per underflow (vuota)	%Ci.E	no
				Uscita preimpostata raggiunta	%Ci.D	no
				Uscita per overflow (piena)	%Ci.F	no

Principi di programmazione dei blocchi funzione standard

Introduzione

Utilizzare uno dei seguenti metodi per programmare i blocchi funzione standard:

- Istruzioni di blocco funzione (ad esempio, `BLK` %TM2): Questo metodo di programmazione in linguaggio Ladder consente di eseguire operazioni sul blocco in una singola posizione del programma.
- Istruzioni specifiche (ad esempio, `CU %Ci`): Questo metodo non reversibile consente di eseguire le operazioni negli ingressi del blocco in posizioni diverse del programma (ad esempio, `riga 100 CU %C1`, `riga 174 CD %C1`, `riga 209 LD %C1.D`).

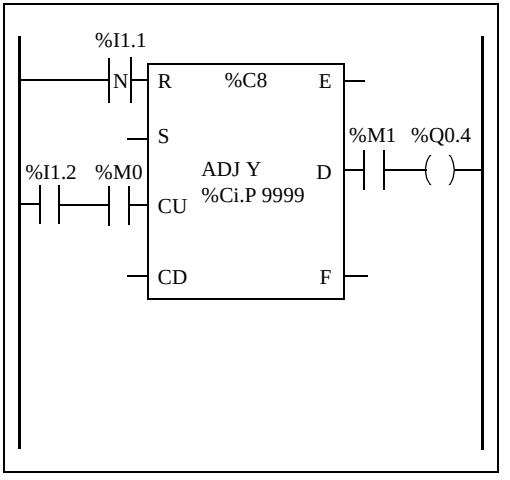
Programmazione reversibile

Utilizzare le istruzioni BLK, OUT_BLK ed END_BLK per la programmazione reversibile:

- **BLK :** Indica l'inizio del blocco.
- **OUT_BLK:** Utilizzato per collegare direttamente le uscite del blocco.
- **END_BLK:** Indica la fine del blocco.

Esempio di uscite collegate

L'esempio che segue mostra la programmazione reversibile di un blocco funzione contatore con uscite collegate.



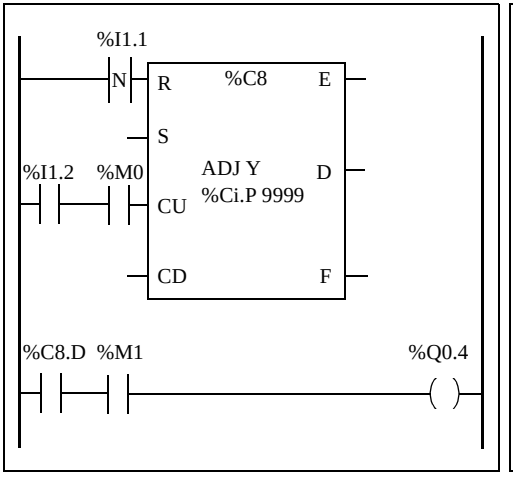

```
BLK     %C8
LDF     %I1.1
R
LD      %I1.2
AND     %M0
CU
OUT_BLK
LD      D
AND     %M1
ST      %Q0.4
END_BLK
```

Elaborazione d'ingresso

Elaborazione d'uscita

Esempi senza collegamento dell'uscita

Questo esempio mostra la programmazione reversibile di un blocco funzione contatore senza uscite collegate.




```
BLK     %C8
LDF     %I1.1
R                  Elaborazione
LD      %I1.2      d'ingresso
AND     %M0
CU
END_BLK
LD      %C8.D
AND     %M1        Elaborazione
ST      %Q0.4      d'uscita
```

Nota: Solo le istruzioni di test e ingresso sul blocco relativo possono essere poste tra le istruzioni BLK e OUT_BLK (o tra BLK ed END_BLK se OUT_BLK non è programmato).

Blocco funzione timer (%TMi)

Introduzione

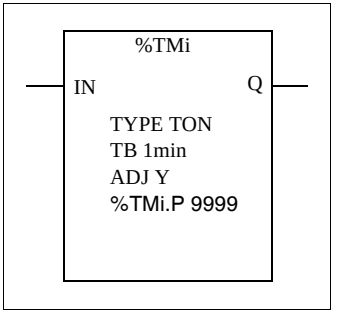
Esistono tre tipi di blocchi funzione Timer:

- TON (timer di ritardo all'eccitazione): utilizzare questo tipo di timer per gestire i ritardi all'eccitazione.
- TOF (timer di ritardo alla diseccitazione): utilizzare questo tipo di timer per gestire i ritardi alla diseccitazione.
- TP (Timer – Pulse, Impulso timer): utilizzare questo tipo di timer per creare un impulso di durata precisa.

I ritardi o gli impulsi sono programmabili e possono essere modificati tramite TwidoSoft.

Illustrazione

Di seguito viene illustrato il blocco funzione Timer.



Blocco funzione timer

Impostazioni

Il blocco funzione timer include i seguenti parametri:

Parametro	Etichetta	Valore
Numero timer	%TMi	da 0 a 63: TWDLCAA10DRF e TWDLCAA16DRF da 0 a 127 per tutti gli altri controller.
Tipo	TON	• ritardo all'eccitazione (predefinito)
	TOF	• ritardo alla diseccitazione
	TP	• pulse (monostabile)
Base tempo	TB	1 min (predefinito), 1 s, 100 ms, 10 ms, 1 ms
Valore corrente	%TMi.V	Parola che aumenta da 0 a %TMi.P quando il timer è in funzione. Può essere letta e provata, ma non scritta dal programma. %TMi.V può essere modificato con l'Editor delle tabelle di animazione.
Valore preimpostato	%TMi.P	0 - 9999. Parola che può essere letta, provata e scritta dal programma. Il valore di default è 9999. Il periodo o ritardo generato è uguale a %TMi.P x TB.
Editor delle tabelle di animazione	Y/N	Y: Sì, il valore %TMi.P preimpostato può essere modificato con l'Editor delle tabelle di animazione. N: No, il valore %TMi.P preimpostato non può essere modificato.
Ingresso convalida (o dell'istruzione)	IN	Avvia il timer sul fronte di salita (tipi TON o TP) o di discesa (tipo TOF).
Uscita timer	Q	Il bit associato %TMi.Q viene impostato a 1 in base alla funzione eseguita: TON, TOF o TP

Nota: La dimensione del valore preimpostato è proporzionale alla precisione del timer.

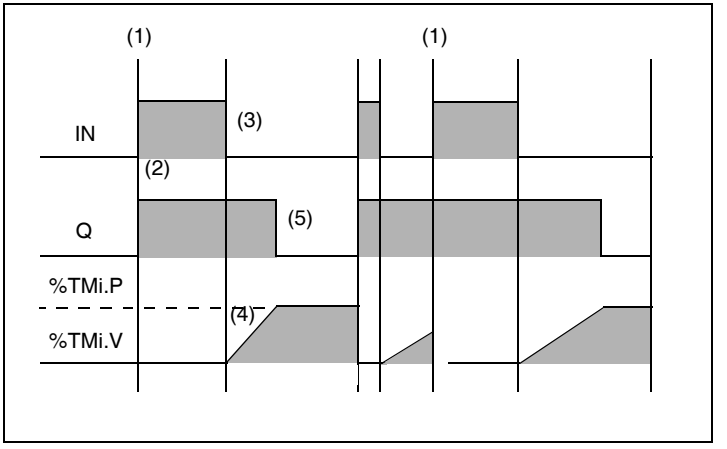
Timer di tipo TOF

Introduzione

Utilizzare il tipo di timer TOF (Timer Off-Delay, Timer di ritardo in OFF) per controllare le azioni di ritardo in OFF. Questo ritardo è programmabile con TwidoSoft.

Diagramma dei tempi

Il seguente diagramma dei tempi illustra il funzionamento del timer di tipo TOF.



Funzionamento

La tabella che segue descrive il funzionamento del timer di tipo TOF.

Fase	Descrizione
1	Il valore corrente di %TMi.V viene impostato a 0 su un fronte di salita all'ingresso IN, anche se il timer è in funzione.
2	Il bit di uscita %TMi.Q viene impostato a 1 quando viene rilevato un fronte di salita all'ingresso N.
3	Il timer si avvia sul fronte di discesa dell'ingresso IN.
4	Il valore corrente %TMi.V aumenta a %TMi.P con incrementi di una unità per ogni impulso della base tempo TB.
5	Il bit di uscita %TMi.Q viene azzerato quando il valore corrente raggiunge %TMi.P.

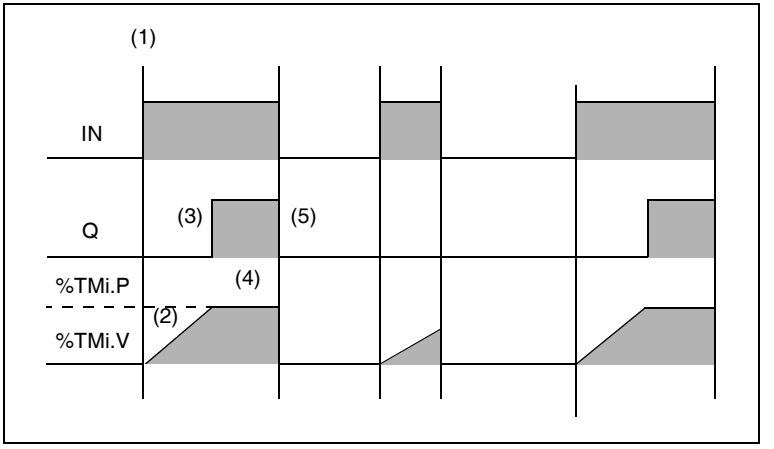
Timer di tipo TON

Introduzione

Il tipo di timer TON (Timer On-Delay, Timer di ritardo in ON) viene utilizzato per controllare le azioni di ritardo in ON. Questo ritardo è programmabile con TwidoSoft.

Diagramma dei tempi

Il seguente diagramma dei tempi illustra il funzionamento del timer di tipo TON.



Funzionamento

La tabella che segue descrive il funzionamento del timer di tipo TON.

Fase	Descrizione
1	Il timer si avvia sul fronte di salita dell'ingresso IN.
2	Il valore corrente %TMi.V aumenta da 0 a %TMi.P con incrementi di una unità per ogni impulso della base tempo TB.
3	Il bit di uscita %TMi.Q viene impostato a 1 quando il valore corrente raggiunge %TMi.P.
4	Il bit di uscita %TMi.Q rimane a 1 mentre l'ingresso IN è a 1.
5	Quando viene rilevato un fronte di discesa all'ingresso IN, il timer viene arrestato, anche se non ha raggiunto %TMi.P e %TMi.V viene impostato a 0.

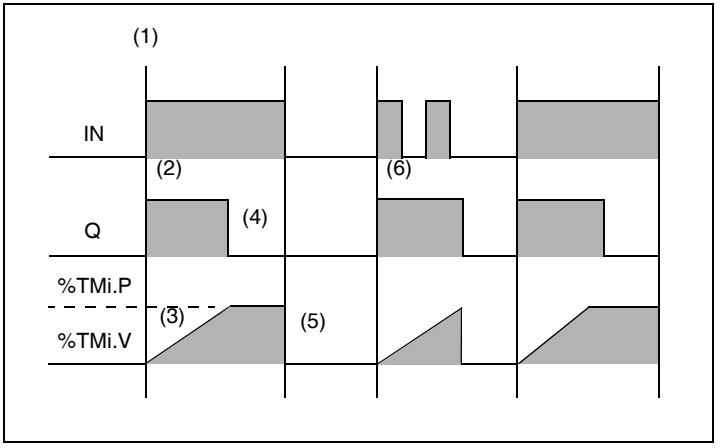
Tipo di timer TP

Introduzione

Il tipo di timer TP (Timer – Pulse, Timer – Impulso) permette di generare impulsi di una durata specifica. TwidoSoft permette di programmare questa durata.

Diagramma dei tempi

Il seguente diagramma dei tempi illustra il funzionamento del timer di tipo TP.



Funzionamento

La tabella che segue descrive il funzionamento del timer di tipo TP.

Fase	Descrizione
1	Il timer si avvia sul fronte di salita dell'ingresso IN. Il valore corrente %TMi.V è impostato a 0 se il timer non si è ancora avviato.
2	Il bit di uscita %TMi.Q viene impostato a 1 all'avvio del timer.
3	Il valore corrente %TMi.V del timer aumenta da 0 a %TMi.P con incrementi di una unità per ogni impulso della base tempo TB.
4	Il bit di uscita %TMi.Q è impostato a 0 quando il valore corrente raggiunge %TMi.P.
5	Il valore corrente %TMi.V viene azzerato quando %TMi.V è uguale a %TMi.P e l'ingresso IN ritorna a 0.
6	Non è possibile reimpostare il timer a zero. Quando %TMi.V è uguale a %TMi.P e l'ingresso IN è impostato a 0, %TMi.V è impostato a 0.

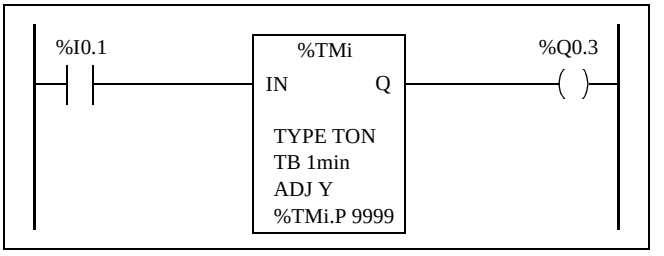
Programmazione e configurazione timer

Introduzione

I blocchi funzione del temporizzatore (Timer) (%TMi) vengono programmati alla stesso modo, indipendentemente da come verranno utilizzati. La funzione timer (TON, TOF o TP) viene selezionata durante la configurazione.

Esempi

L'illustrazione che segue mostra un blocco funzione timer con esempi di programmazione reversibile e non reversibile.



Programmazione reversibile

```
BLK     %TM1
LD      %I0.1
IN
OUT_BLK
LD      Q
ST      %Q0.3
END_BLK
```

Programmazione non reversibile

```
LD      %I0.1
IN      %TM1
LD      %TM1.Q
ST      %Q0.3
```

Configurazione

Durante la configurazione devono essere specificati i seguenti parametri:

- Tipo di temporizzatore: TON, TOF o TP
- Base tempo (TB): 1 min, 1 s, 100 ms, 10 ms o 1 ms
- Valore di preselezione (%TMi.P): da 0 a 9999
- Regolazione: Contrassegnato o non contrassegnato

Casi speciali

La tabella seguente contiene un elenco dei casi speciali di programmazione del blocco funzione Timer.

Caso speciale	Descrizione
Effetto di un avvio a freddo (%S0=1)	Azzera il valore corrente. Imposta l'uscita %TMi.Q a 0. Il valore preimpostato viene ripristinato al valore definito durante la configurazione.
Effetto di un riavvio a caldo (%S1=1)	Nessun effetto sui valori correnti e preimpostati del timer. Il valore corrente non cambia in caso di interruzione dell'alimentazione.
Effetto di un arresto del controller	L'arresto del controller non blocca il valore corrente.
Effetto di un salto di programma	Il salto del blocco timer non provoca il blocco del timer stesso. Il timer continua a incrementare fino a raggiungere il valore preimpostato (%TMi.P). A questo punto, lo stato del bit Completato (%TMi.Q) visualizzato all'uscita Q del blocco timer viene modificato. L'uscita associata, tuttavia, collegata direttamente all'uscita del blocco, non viene né attivata né scansionata dal controller.
Prova con il bit %TMi.Q (bit di conferma)	È opportuno provare il bit %TMi.Q solo una volta nel programma.
Effetto della modifica di %TMi.P preimpostato	La modifica del valore preimpostato tramite un'istruzione o regolando il valore ha effetto solo dalla successiva attivazione del timer.

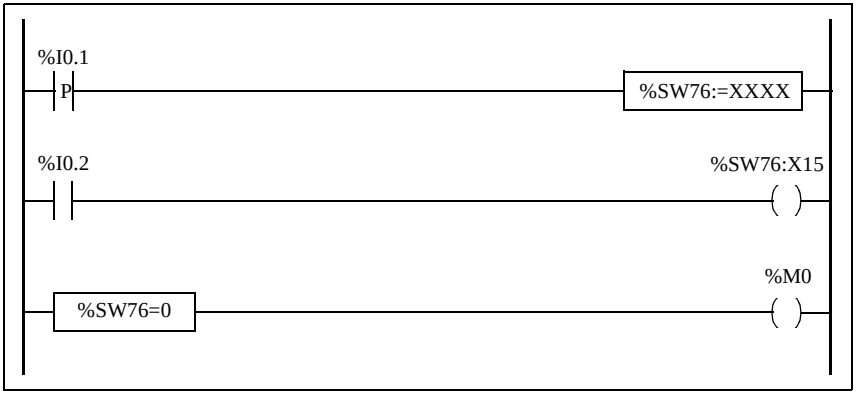
Timer con base tempo di 1 ms

La base tempo di 1 ms è disponibile solo con i primi cinque timer. Le quattro parole di sistema %SW76, %SW77, %SW78, and SW79 possono essere usate come "clessidre". Vengono decrementate singolarmente dal sistema ogni millisecondo **se hanno un valore positivo.**
È possibile ottenere una temporizzazione multipla tramite caricamenti successivi di queste parole o provando i valori intermedi. Se il valore di una di queste quattro parole è inferiore a 0, la parola non verrà modificata. Un timer può essere bloccato impostando il bit 15 corrispondente a 1 e "sbloccato" azzerando lo stesso bit.

Esempio di programmazione

Di seguito è riportato un esempio di programmazione di un blocco funzione del timer.

```
LDR     %I0.1          (Avvio del timer sul fronte di salita di %I0.1)
[%SW76:=XXXX]          (XXXX = valore richiesto)
LD      %I0.2          (gestione facoltativa di blocco, l'ingresso I0.2 si blocca)
ST      %SW76:X15
LD      [%SW76=0]      (test fine timer)
ST      %M0
.............
```



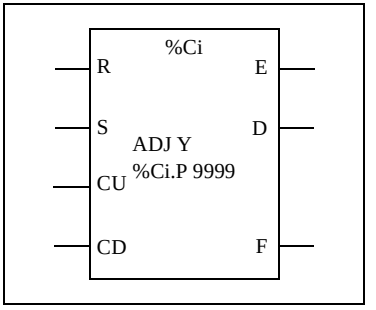
Blocco funzione contatore avanti/indietro (%Ci)

Introduzione

Il blocco funzione contatore (%Ci) consente il conteggio avanti/indietro degli eventi. Queste due operazioni possono essere eseguite contemporaneamente.

Illustrazione

Di seguito viene illustrato il blocco funzione Contatore avanti/indietro.



Blocco funzione contatore avanti/indietro

Impostazioni

Il blocco funzione Contatore presenta i seguenti parametri:

Parametro	Etichetta	Valore
Numero contatore	%Ci	0 à 127
Valore corrente	%Ci.V	Il valore della parola aumenta o diminuisce di un'unità in base agli ingressi (o istruzioni) CU e CD. Può essere letta e provata, ma non scritta dal programma. Utilizzare l'Editor di dati per modificare %Ci.V.
Valore preimpostato	%Ci.P	0 ≤ %Ci.P ≤ 9999. La parola può essere letta, provata e scritta (valore predefinito: 9999).
Modifica tramite Editor delle tabelle di animazione	ADJ	• Y: Sì, il valore preimpostato può essere modificato con l'Editor delle tabelle di animazione. • N: No, il valore preimpostato non può essere modificato con l'Editor delle tabelle di animazione.
Ingresso (o istruzione) Reset	R	Allo stato 1: %Ci.V = 0.
Ingresso (o istruzione) di preselezione	S	Allo stato 1: %Ci.V = %Ci.P.
Ingresso (o istruzione) di conteggio in avanti	CU	Aumenta il valore %Ci.V di un'unità su un fronte di salita.

Parametro	Etichetta	Valore
Ingresso (o istruzione) di conteggio in indietro	CD	Diminuisce il valore %Ci.V di un'unità su un fronte di salita.
Uscita overflow conteggio indietro	E (Empty, Vuoto)	Il bit associato %Ci.E=1, quando il contatore indietro %Ci.V passa da 0 a 9999 (impostato a 1 quando %Ci.V raggiunge 9999 e azzerato se il contatore continua il conteggio indietro).
Uscita preimpostata raggiunta	D (Done, Conferma)	Il bit associato %Ci.D=1, quando %Ci.V=%Ci.P.
Uscita overflow conteggio avanti	F (Full)	Il bit associato %Ci.F=1, quando %Ci.V passa da 9999 a 0 (impostato a 1 quando %Ci.V raggiunge 0 e azzerato se il contatore continua il conteggio in avanti).

Funzionamento La tabella che segue descrive le fasi principali del funzionamento del contatore avanti/indietro.

Funzionamento	Azione	Risultato
Conteggio in avanti	Appare un fronte di salita all'ingresso CU di conteggio (oppure viene abilitata l'istruzione CU).	Il valore corrente di %Ci.V aumenta di una unità.
	Il valore corrente di %Ci.V è uguale al valore preimpostato di %Ci.P.	Il bit di uscita "preset. raggiunto" %Ci.D passa a 1.
	Il valore corrente di %Ci.V passa da 9999 a 0.	Il bit di uscita %Ci.F (overflow conteggio) passa a 1.
	Se il contatore continua il conteggio.	Il bit di uscita %Ci.F (overflow conteggio) torna a 0.
Conteggio indietro	Appare un fronte di salita all'ingresso CD di conteggio indietro (oppure viene abilitata l'istruzione CD).	Il valore corrente %Ci.V diminuisce di una unità.
	Il valore corrente di %Ci.V passa da 0 a 9999.	Il bit di uscita %Ci.E (overflow conteggio indietro) passa a 1.
	Se il contatore continua il conteggio indietro.	Il bit di uscita %Ci.F (overflow conteggio indietro) torna a 0.
Conteggio avanti/ indietro	Per utilizzare contemporaneamente le funzioni di conteggio avanti e indietro (o per abilitare le istruzioni CD e CU), occorre comandare contemporaneamente i due ingressi corrispondenti CU e CD. Tali ingressi vengono quindi sottoposti a scansione. Se il valore = 1, il valore corrente non viene modificato.	
Ripristino	Impostazione a 1 dell'ingresso R (oppure viene abilitata l'istruzione R).	Forza l'azzeramento del valore %Ci.V. Le uscite %Ci.E, %Ci.D e %Ci.F sono impostate a 0. L'azzeramento dell'ingresso è prioritario.

Funzionamento	Azione	Risultato
Preselezione	Se si imposta a 1 l'ingresso S (o se si abilita l'istruzione S) e se l'ingresso di azzeramento è allo stato 0 (o se l'istruzione R è inattiva).	Il valore corrente %Ci.V assume il valore di %Ci.P e l'uscita %Ci.D viene impostata a 1.

Casi speciali

La tabella seguente contiene un elenco di casi speciali di funzionamento e configurazione dei contatori.

Caso speciale	Descrizione
Effetto di un riavvio a freddo (%S0=1)	• Il valore corrente di %Ci.V è azzerato. • I bit di uscita %Ci.E, %Ci.D e %Ci.F sono impostati a 0. • Il valore preimpostato viene inizializzato con il valore definito durante la configurazione.
Effetto di un riavvio a caldo (%S1=1) di un arresto del controller (STOP)	Non ha effetto sul valore corrente del contatore (%Ci.V).
Effetto della modifica di %Ci.P preimpostato	La modifica del valore preimpostato tramite un'istruzione o regolazione ha effetto quando il blocco viene elaborato dall'applicazione (abilitazione di uno degli ingressi).

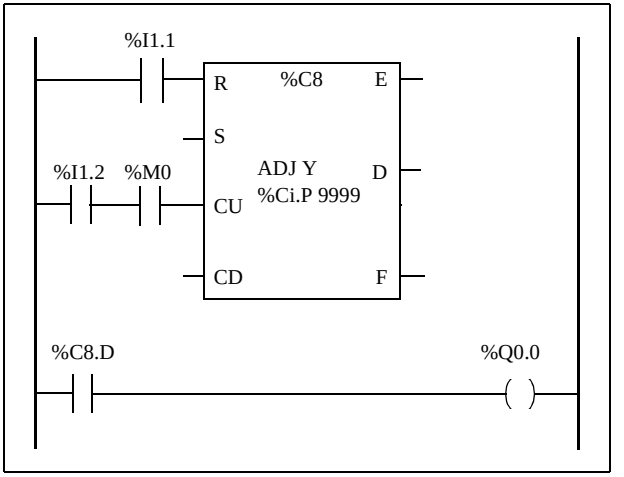
Programmazione e configurazione contatori

Introduzione

L'esempio seguente mostra un contatore che consente di contare fino a 5.000 articoli. Ogni impulso all'ingresso %I1.2 (quando il bit interno %M0 è a 1) incrementa il valore del contatore %C8 di una unità, fino al valore di preselezione finale (bit %C8.D=1). Il contatore viene azzerato dall'ingresso %I1.1.

Esempio di programmazione

L'illustrazione che segue mostra un blocco funzione contatore con esempi di programmazione reversibile e non reversibile.



Schema Ladder

```
BLK     %C8
LD      %I1.1
R
LD      %I1.2
AND     %M0
CU
END_BLK
LD      %C8.D
ST      %Q0.0
```

Programmazione reversibile

```
LD      %I1.1
R       %C8
LD      %I1.2
AND     %M0
CU      %C8
LD      %C8.D
ST      %Q0.0
```

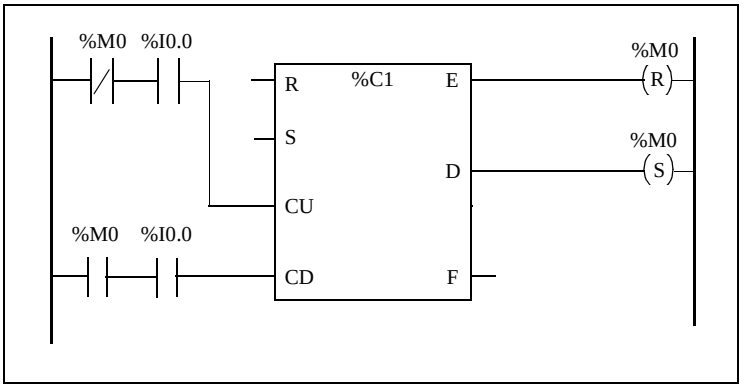
Programmazione non reversibile

Configurazione

Durante la configurazione devono essere specificati i seguenti parametri:
- Valore di preselezione (%Ci.P): per questo esempio fissato a 5000
- Regolazione: Sì

Esempio di un contatore avanti/ indietro

Di seguito viene illustrato un blocco funzione Contatore avanti/indietro.



Schema Ladder

In questo esempio, se si prende %C1.P 4, il valore corrente del contatore %C1.V verrà incrementato da 0 a 3, quindi decrementato da 3 a 0. Finché %I0.0=1, %C1.V oscilla tra 0 e 3.

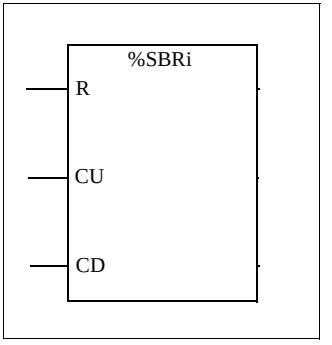
Blocco funzione registro bit di scorrimento (%SBRi)

Introduzione

Il blocco funzione registro bit di scorrimento (%SBRi) attua uno scorrimento a sinistra o destra dei bit di dati binari (0 o 1).

Illustrazione

Di seguito è riportato un esempio di blocco funzione registro di scorrimento.



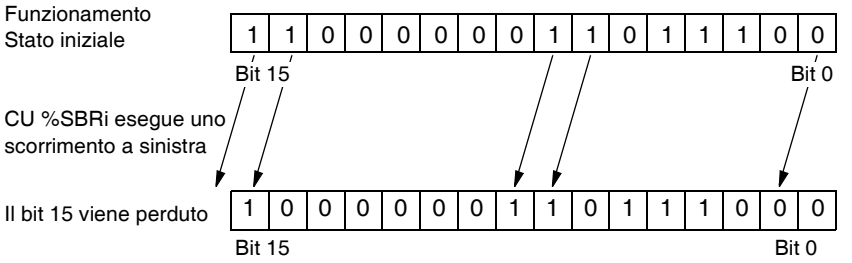
Impostazioni

Il blocco funzione registro bit di scorrimento presenta i seguenti parametri:

Parametro	Etichetta	Valore
Numero registro	%SBRi	da 0 a 7
Bit di registro	%SBRi.j	I bit da 0 a 15 (j = 0 - 15) del registro di scorrimento possono essere provati da un'istruzione di test e scritti con un'istruzione di assegnazione.
Azzera ingresso (o istruzione)	R	Quando il parametro di funzione R è 1, i bit di registro da 0 a 15 %SBRi.j vengono impostati a 0.
Scorre a sinistra l'ingresso (o l'istruzione)	CU	In un fronte di salita, fa scorrere un bit di registro a sinistra.
Scorre a destra l'ingresso (o l'istruzione)	CD	In un fronte di salita, fa scorrere un bit di registro a destra.

Funzionamento

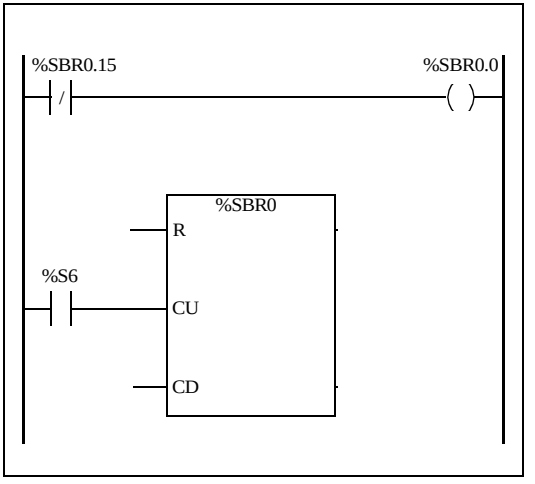
L'illustrazione che segue mostra un pattern di bit prima e dopo l'operazione di scorrimento.



Corrisponde anche alla richiesta di scorrimento di un bit a destra (da bit 15 a bit 0) con l'istruzione CD. Il bit 0 viene perduto.
Se un registro a 16 bit non è adeguato, è possibile utilizzare il programma per sovrapporre più registri.

Programmazione

Nell'esempio che segue, si fa scorrere a sinistra un bit al secondo mentre il bit 0 assume lo stato opposto rispetto al bit 15.



Programmazione reversibile

```
LDN    %SBR0.15
ST     %SBR0.0
BLK    %SBR0
LD     %S6
CU
END_BLK
```

Programmazione reversibile

```
LDN    %SBR0.15
ST     %SBR0.0
LD     %S6
CU     %SBR0
```

Casi speciali

La tabella seguente contiene un elenco dei casi speciali di programmazione del blocco funzione Registro bit di scorrimento.

Caso speciale	Descrizione
Effetto di un avvio a freddo (%S0=1)	Imposta tutti i bit della parola di registro a 0.
Effetto di un riavvio a caldo (%S1=1)	Non ha effetto sui bit della parola di registro.

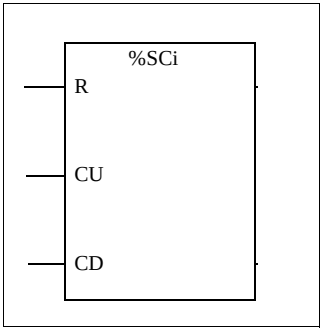
Blocco funzione contatore passo passo (%SCi)

Introduzione

Il blocco funzione contatore passo passo (%SCi) fornisce una serie di passi a cui è possibile assegnare le azioni. Il passaggio da un passo a un altro dipende da eventi interni o esterni. Ogni volta che un passo è attivo, il bit associato è impostato a 1. Può essere attivo solo un passo di un contatore passo passo alla volta.

Illustrazione

Di seguito è riportato un esempio di un blocco funzione contatore passo passo.



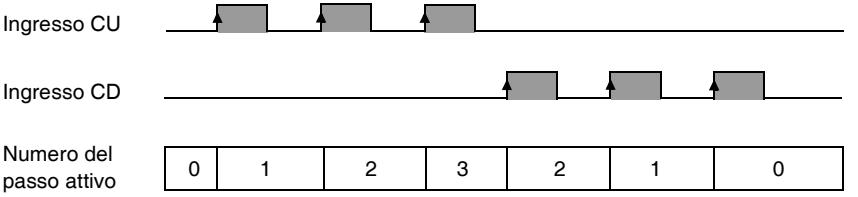
Parametri

Il blocco funzione contatore passo passo presenta i seguenti parametri:

Parametro	Etichetta	Valore
Numero contatore passo passo	%SCi	0 - 7
Bit contatore passo passo	%SCi.j	I bit del contatore passo passo da 0 a 255 (j = da 0 a 255) possono essere provati tramite un'operazione logica Load e scritti da un'istruzione di assegnazione.
Azzera ingresso (o istruzione)	R	Quando il parametro di funzione R è 1, il contatore passo passo viene azzerato.
Incrementa ingresso (o istruzione)	CU	In un fronte di salita, incrementa il contatore di un passo.
Decrementa ingresso (o istruzione)	CD	In un fronte di salita, diminuisce il contatore di un passo.

Diagramma dei tempi

Il diagramma che segue illustra il funzionamento del blocco funzione contatore passo passo.

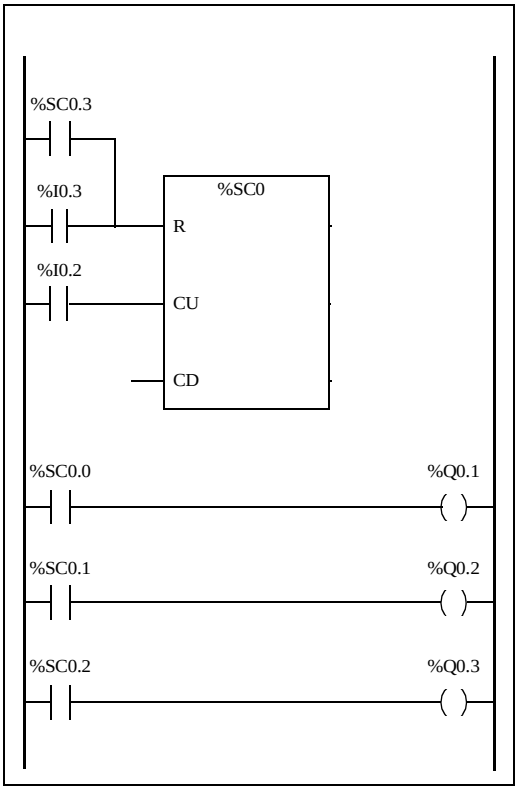


Programmazione

Di seguito è riportato un esempio di un blocco funzione contatore passo passo.

- Il contatore passo passo 0 viene incrementato dall'ingresso %I0.2.
- Il contatore passo passo 0 viene azzerato dall'ingresso %I0.3 oppure quando giunge al passo 3.
- Il passo 0 controlla l'uscita %Q0.1, il passo 1 controlla l'uscita %Q0.2 e il passo 2 controlla l'uscita %Q0.3.

L'illustrazione che segue mostra la programmazione reversibile e non reversibile di questo esempio.



Programmazione reversibile

```
BLK     %SC0
LD      %SC0.3
OR      %I0.3
R
LD      %I0.2
CU
END_BLK
LD      %SC0.0
ST      %Q0.1
LD      %SC0.1
ST      %Q0.2
LD      %SC0.2
```

Programmazione non reversibile

```
LD      %SC0.3
OR      %I0.3
R       %SC0
LD      %I0.2
CU      %SC0
LD      %SC0.0
ST      %Q0.1
LD      %SC0.1
ST      %Q0.2
LD      %SC0.2
```

Caso speciale

La tabella seguente contiene un elenco dei casi speciali riguardanti il funzionamento del blocco funzione contatore passo passo.

Caso speciale	Descrizione
Effetto di un avvio a freddo (%S0=1)	Inizializza il contatore passo passo.
Effetto di un riavvio a caldo (%S1=1)	Non ha effetto sul contatore passo passo.

16.3 Elaborazione numerica

In breve

Oggetto di questa sezione

Questa sezione presenta un'introduzione all'elaborazione numerica, oltre a descrizioni e indicazioni di programmazione.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Introduzione alle istruzioni digitali	416
Istruzioni di assegnazione	417
Istruzioni di confronto	422
Istruzioni aritmetiche su interi	424
Istruzioni logiche	428
Istruzioni di scorrimento	429
Istruzioni di conversione	431
Istruzioni di conversione tra parole semplici e doppie	433

Introduzione alle istruzioni digitali

In breve

Le istruzioni digitali si applicano generalmente alle parole di 16 bit (vedere sezione *Oggetti parola, p. 29*) e alle parole doppie di 32 bit (Vedi *Oggetti a virgola mobile e parola doppia, p. 32*). scritte tra parentesi quadre. Se il risultato dell'operazione logica precedente è vero (accumulatore booleano = 1), l'istruzione numerica viene eseguita. Se il risultato dell'operazione logica precedente è falso (accumulatore booleano = 0), l'istruzione numerica non viene eseguita.

Istruzioni di assegnazione

Introduzione

Le istruzioni di assegnazione vengono utilizzate per caricare l'operando Op2 nell'operando Op1.

Assegnazione

Sintassi delle istruzioni di assegnazione.

[Op1:=Op2]	<=>	Op2 -> Op1

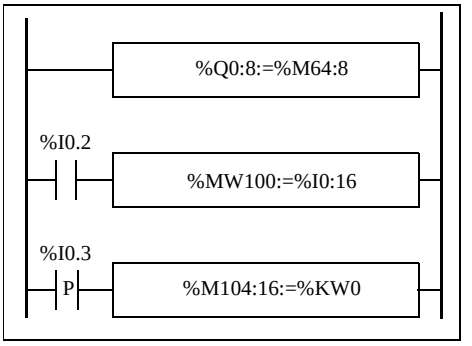
Le operazioni di assegnazione possono essere eseguite su:
- Stringhe di bit
- Parole
- Parole doppie
- Virgola mobile
- Tabelle di parole
- Tabelle di parole doppie
- Tabelle in virgola mobile

Assegnazione di stringhe di bit

Le operazioni possono essere eseguite sulle stringhe di bit seguenti (vedere *Oggetti strutturati, p. 46*):
- Stringa di bit-> stringa di bit (esempio 1)
- Stringa di bit-> parola (esempio 2) o parola doppia (indicizzata)
- Parola o parola doppia (indicizzata) -> stringa di bit (esempio 3)
- Valore immediato -> stringa di bit

Esempi

Esempi di assegnazioni di stringa di bit.




```
LD      1
[%Q0:8:=%M64:8]          (Ex. 1)

LD      %I0.2
[%MW100:=%I0:16]         (Ex. 2)

LDR     %I0.3
[%M104:16:=%KW0]         (Ex. 3)
```

Regole di utilizzo:

- Per l'assegnazione stringa di bit -> parola: i bit nella stringa vengono trasferiti nella parola iniziando da destra (primo bit nella stringa verso il bit 0 nella parola) e i bit della parola non coinvolti nel trasferimento (lunghezza≤16) vengono azzerati.
- Per l'assegnazione parola -> stringa di bit: i bit della parola vengono trasferiti da destra (bit parola 0 verso il primo bit nella stringa).

Assegnazioni stringa di bit

Sintassi delle assegnazioni stringa di bit.

Operatore	Sintassi	Operando 1 (Op1)	Operando 2 (Op2)
:=	[Op1 : = Op2] L'operando 1 (Op1) assume il valore dell'operando 2 (Op2).	%MWi,%QWi, %QWAi,%SWi %MWi[%MWi], %MDi, %MDi[%MWi] **%Mi:L, %Qi:L, %Si:L, %Xi:L**	Valore immediato, %MWi, %KWi, %IW,%IWAi, %INWi, %QWi, %QWAi %QNWi, %SWi, %BLK.x, %MWi[%MWi], %KWi[%MWi], %MDi[%MWi], %KDi[%MWi], **%Mi:L,%Qi:L, %Si:L, %Xi:L, %Ii:L**

Nota: l'abbreviazione %BLK.x (ad esempio, %C0.P) viene utilizzata per descrivere qualsiasi parola del blocco funzione.

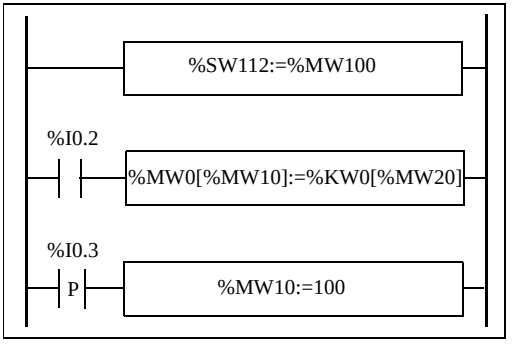
Assegnazione di parole

Le operazioni di assegnazione possono essere eseguite sulle seguenti parole e parole doppie:

- Parola (indicizzata) -> parola (esempio 2) (indicizzata o no)
- Parola doppia (indicizzata) -> Parola doppia (indicizzata o no)
- Valore intero immediato -> parola (esempio 3) o parola doppia (indicizzate o no)
- Stringa di bit-> parola o parola doppia
- Virgola mobile (indicizzato o no)-> virgola mobile (indicizzato o no)
- Parola o parola doppia -> stringa di bit
- Virgola mobile immediata -> virgola mobile (indicizzato o no)

Esempi

Esempi di assegnazioni di parole.




```
LD      1
[%SW112:=%MW100]      (Ex. 1)

LD      %I0.2
[%MW0[%MW10]:=        (Ex. 2)
%KW0[%MW20]]

LDR     %I0.3         (Ex. 3)
[%MW10:=100]
```

Sintassi

Sintassi delle assegnazioni di parole

Operatore	Sintassi
:=	[Op1 : = Op2] L'operando 1 (Op1) assume il valore dell'operando 2 (Op2).

La tabella seguente descrive in modo dettagliato gli operandi:

Tipo	Operando 1 (Op1)	Operando 2 (Op2)
parola, parola doppia, stringa di bit	**%BLK.x, %MWi, %QWi, %QWAi, %SWi %MWi[MWi**, %MDi, %MDi[%MWj]], %Mi:L, %Qi:L, %Si:L, %Xi:L	**Valore immediato, %MWi, %KWi, %IW, %IWAi, %QWi, %QWAi, %SWi, %MWi[MWi], %KWi[MWi], %MDi, %MDi[%MWj], %KDi, %KDi[MWj]** , %INW, %Mi:L, %Qi:L, %QNW, %Si:L, %Xi:L, %Ii:L
Virgola mobile	**%MFi, %MFi[%MWj]**	**valore virgola mobile immediato, %MFi, %MFi[%MWj], %KFi, %KFi[%MWj]**

Nota: l'abbreviazione %BLK.x (ad esempio, %R3.I) viene utilizzata per descrivere qualsiasi parola del blocco funzione. Per le stringhe di bit %Mi:L, %Si:L e %Xi:L, l'indirizzo base del primo bit della stringa deve essere un multiplo di 8 (0, 8, 16, ..., 96, ...).

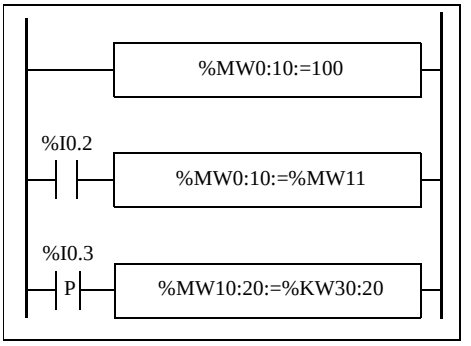
Assegnazione di tabelle di parole, parole doppie o virgole mobili

Le operazioni di assegnazione possono essere eseguite sulle seguenti tabelle di oggetti (vedere *Tabelle di parole, p. 47*):

- Valore intero immediato -> tabella di parole (esempio 1) o di parole doppie
- Parola -> tabella di parole (esempio 2)
- Tabella di parole -> tabella di parole (esempio 3)
  La lunghezza della tabella (L) deve essere uguale per entrambe le tabelle.
- Parola doppia -> tabella di parole doppie
- Tabella di parole doppie -> tabella di parole doppie
  La lunghezza della tabella (L) deve essere uguale per entrambe le tabelle.
- Valore a virgola mobile immediato -> tabella a virgola mobile
- Virgola mobile -> tabella a virgola mobile
- Tabella a virgola mobile -> tabella a virgola mobile
  La lunghezza della tabella (L) deve essere uguale per entrambe le tabelle.

Esempi

Esempi di assegnazioni di tabelle di parole.




```
LD      1
[%MW0:10:=100]                (Ex. 1)

LD      %I0.2
[%MW0:10:=%MW11]              (Ex. 2)

LDR     %I0.3
[%MW10:20:=%KW30:20]          (Ex. 3)
```

Sintassi

Sintassi delle assegnazioni di tabelle di parole, parole doppie e virgole mobili

Operatore	Sintassi
:=	[Op1 : = Op2] L'operando 1 (Op1) assume il valore dell'operando 2 (Op2).

La tabella seguente descrive in modo dettagliato gli operandi:

Tipo	Operando 1 (Op1)	Operando 2 (Op2)
tabella di parole	**%MWi:L, %SWi:L**	**%MWi:L, %SWi:L**, Valore intero immediato, %MWi, %KWi, %IW, %QW, %IWA, %QWA, %SWi, %BLK.x
tabelle di parole doppie	**%MDi:L**	**valore intero immediato, %MDi, %KDi,%MDi:L, %KDi:L**
tabella di virgola mobile	**%MFi:L]**	**valore virgola mobile immediato, %MFi, %KFi, %MFi:L, %KFi:L**

Nota: l'abbreviazione %BLK.x (ad esempio, %R3.I) viene utilizzata per descrivere qualsiasi parola del blocco funzione.

Istruzioni di confronto

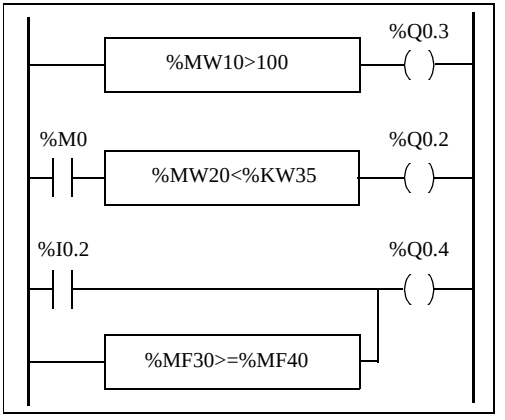
Introduzione

Le istruzioni di confronto vengono utilizzate per confrontare due operandi.
La tabella che segue elenca i tipi di istruzioni di confronto.

Istruzione	Funzione
>	Prova se l'operando 1 è maggiore dell'operando 2.
>=	Prova se l'operando 1 è maggiore o uguale all'operando 2.
<	Prova se l'operando 1 è minore dell'operando 2.
<=	Prova se l'operando 1 è minore o uguale all'operando 2.
=	Prova se l'operando 1 è uguale all'operando 2.
<>	Prova se l'operando 1 è diverso dall'operando 2.

Struttura

Il confronto avviene tra le parentesi quadre che seguono le istruzioni LD, AND e OR.
Il risultato è 1 se il confronto richiesto risulta Vero.
Esempi di istruzioni di confronto.




```
LD    [%MW10 > 100]
ST    %Q0.3

LD    %M0
AND   [%MW20 < %KW35]
ST    %Q0.2

LD    %I0.2
OR    [%MF30>=%MF40]
ST    %Q0.4
```

Sintassi

Sintassi delle istruzioni di confronto:

Operatore	Sintassi
>, >=, <, <=, =, <>	LD [Op1 Operatore Op2] AND [Op1 Operatore Op2] OR [Op1 Operatore Op2]

Operandi:

Tipo	Operando 1 (Op1)	Operando 2 (Op2)
Parole	%MWi, %KWi, %INWi, %IW, %IWAi, %QNWi, %QWi, %QWAi, %QNWi, %SWi, %BLK.x	Valore immediato, %MWi, %KWi, %INWi, %IW, %IWAi, %QNWi, %QW, %QWAi, %SWi, %BLK.x, %MWi [%MWi], %KWi [%MWi]
Parole doppie	%MDi, %KDi	Valore immediato, %MDi, %KDi, %MDi [%MWi], %KD [%MWi]
Virgola mobile	%MFi, %KFi	Valore immediato in virgola mobile, %MFi, %KFi, %MFi [%MWi], %KFi [%MWi]

Nota: Le istruzioni di confronto possono essere utilizzate tra parentesi.

Esempio di utilizzo dell'istruzione di confronto tra parentesi:

```
LD      %M0
AND(    [%MF20 > 10.0]
OR      %I0.0
)
ST      %Q0.1
```

Istruzioni aritmetiche su interi

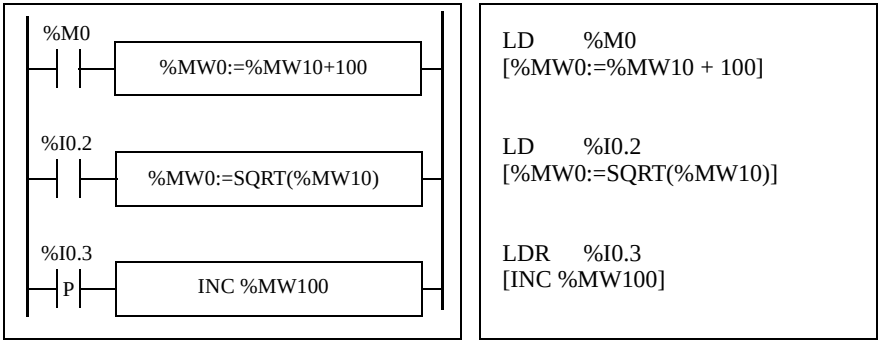
Introduzione

Le istruzioni aritmetiche vengono utilizzate per eseguire operazioni aritmetiche tra due operandi o su un operando intero.
La tabella che segue elenca i tipi di istruzioni aritmetiche.

Istruzione	Funzione
+	Somma due operandi
-	Sottrae due operandi
*	Moltiplica due operandi
/	Divide due operandi
REM	Resto della divisione dei due operandi
SQRT	Radice quadrata di un operando
INC	Incrementa un operando
DEC	Decrementa un operando
ABS	Valore assoluto di un operando

Struttura

Le operazioni aritmetiche vengono eseguite nel modo che segue:



Sintassi

La sintassi dipende dagli operatori utilizzati, come indicato nella tabella di seguito.

Operatore	**Sintassi**
+, -, *, /, REM	[Op1: = Op 2 Operatore Op3]
INC, DEC	[Operatore Op1]
SQRT (1)	[Op1: = SQRT(Op2)]
ABS (1)	[Op1: = ABS(Op2)]

Operandi:

Tipo	**Operando 1 (Op1)**	**Operandi 2 e 3 (Op2 & 3) (1)**
Parole	%MWi, %QWi, %QWAi, %SWi	Valore immediato, %MWi, %KWi, %INW, %IW, %IWAi, %QNW, %QW, %QWAi, %SWi, %BLK.x
Parole doppie	%MDi	Valore immediato, %MDi, %KDi

Nota: (1) con questo operatore, Op2 non può essere un valore immediato.
La funzione ABS è utilizzabile solo con parole doppie (%MD e %KD) e valori a virgola mobile (%MF e %KF). Di conseguenza, OP1 e OP2 devono essere parole doppie o valori a virgola mobile.

Condizioni di errore e overflow

Addizione

- Overflow durante l'operazione su parole
  Se il risultato supera la capacità della parola del risultato, il bit %S18 (overflow) viene impostato a 1 e il risultato non è quindi significativo (vedere l'esempio 1 nella pagina successiva). Il programma utente gestisce il bit %S18.

Nota:
per le parole doppie, i limiti disponibili sono -2147483648 e 21474836487.

Moltiplicazione

- Overflow durante l'operazione
  Se il risultato supera la capacità della parola del risultato, il bit %S18 (overflow) viene impostato a 1 e il risultato non è significativo.

Divisione / resto

- Divisione per 0
  Se il divisore è 0, la divisione è impossibile e il bit di sistema %S18 viene impostato a 1. Il risultato non è corretto.
- Overflow durante l'operazione
  Se il quoziente della divisione eccede la capacità della parola del risultato, il bit %S18 viene impostato a 1.

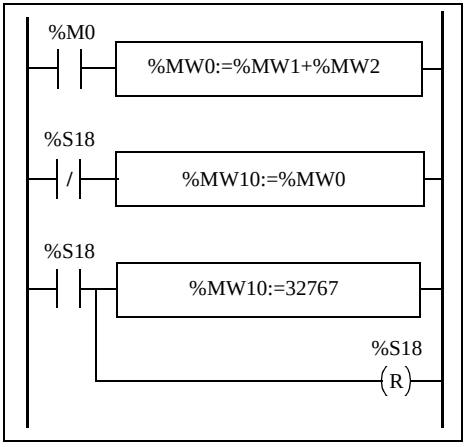
Estrazione della radice quadrata

- Overflow durante l'operazione
  L'estrazione della radice quadrata viene eseguita solo con i valori positivi. Il risultato, quindi, è sempre positivo. Se l'operando della radice quadrata è negativo, il bit di sistema %S18 viene impostato a 1 e il risultato non è corretto.

Nota: il programma utente gestisce i bit di sistema %S17 e %S18. Questi bit sono impostati a 1 dal controller e devono essere reimpostati dal programma per poter essere riutilizzati (vedere l'esempio nella pagina precedente).

Esempi

Esempio 1: overflow durante l'addizione




```
LD      %M0
[%MW0:=%MW1 + %MW2]

LDN     %S18
[%MW10:=%MW0]

LD      %S18
[%MW10:=32767]
R       %S18
```

Se %MW1 =23241 e %MW2=21853, il risultato reale (45094) non può essere espresso in una parola a 16 bit, il bit %S18 viene impostato a 1 e il risultato ottenuto (-20442) non è corretto. In questo esempio se il risultato è maggiore di 32767, il valore viene fissato a 32767.

Istruzioni logiche

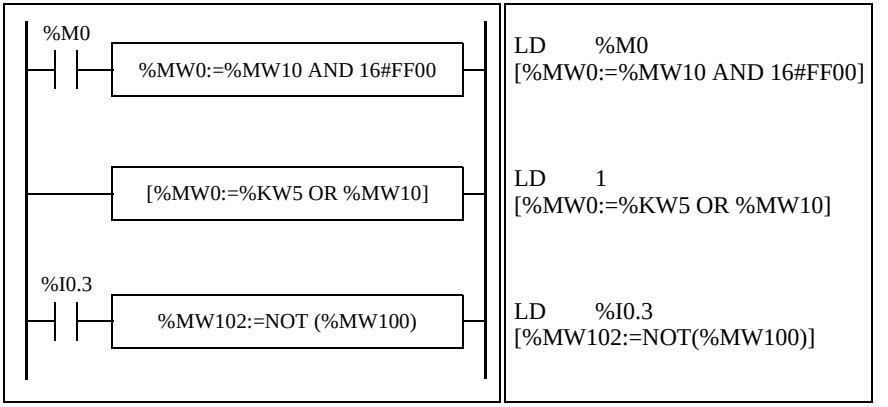
Introduzione

Le istruzioni logiche vengono utilizzate per eseguire operazioni logiche tra due operandi o su un operando.
La tabella che segue elenca i diversi tipi di istruzioni logiche:

Istruzione	Funzione
AND	AND (bit a bit) tra due operandi
OR	OR logico (bit a bit) tra due operandi
XOR	OR esclusivo (bit a bit) tra due operandi
NOT	Complemento logico (bit a bit) di un operando

Struttura

Le operazioni logiche vengono eseguite nel modo che segue:



Sintassi

La sintassi dipende dall'operatore utilizzato:

Operatore	Sintassi	Operando 1 (Op1)	Operandi 2 e 3 (Op2 e 3)
AND, OR, XOR	[Op1: = Op2 Operatore Op3]	%MWi, %QWi, %QWAi, %SWi	Valore immediato (1), %MWi, %KWi, %IW, %IWAi, %QW, %QWAi, %SWi, %BLK.x
NOT	[Op1:=NOT(Op2)]		

Nota: (1) con NOT, Op2 non può essere un valore immediato.

Esempio

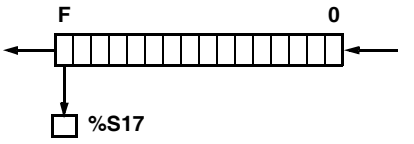
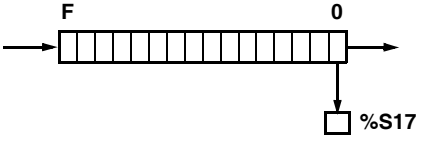
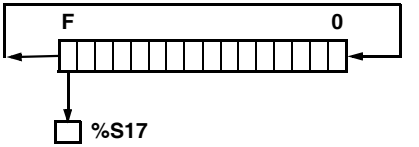
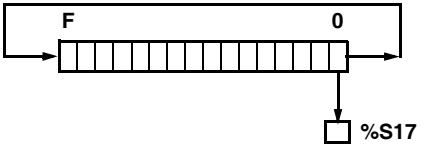
Di seguito è riportato un esempio dell'istruzione AND logico.

```
[%MW15:=%MW32 AND %MW12]
```

Istruzioni di scorrimento

Introduzione

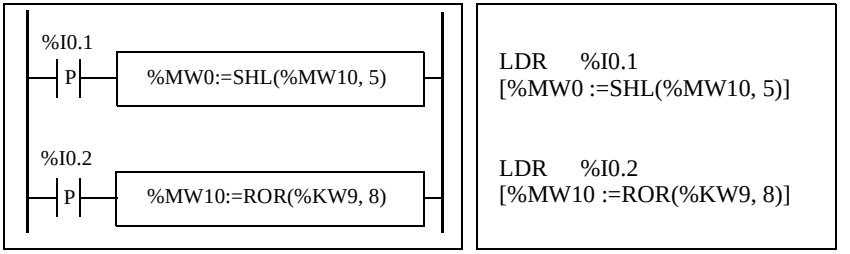
Le istruzioni di scorrimento spostano i bit di un operando di un determinato numero di posizioni a destra o a sinistra.
La tabella che segue elenca i tipi di istruzioni di scorrimento.

Istruzione	Funzione	
Scorrimento logico		
SHL(op2,i)	Scorrimento logico di i posizioni a sinistra	
SHR(op2,i)	Scorrimento logico di i posizioni a destra	
Scorrimento circolare		
ROL(op2,i)	Scorrimento circolare di i posizioni a sinistra	
ROL(op2,i)	Scorrimento circolare di i posizioni a destra	

Nota: il bit di sistema %S17 (Vedi *Bit di sistema (%S), p. 604*) viene utilizzato per il superamento di capacità.

Struttura

Le operazioni di scorrimento vengono eseguite nel modo che segue:



Sintassi

La sintassi dipende dagli operatori utilizzati, come indicato nella tabella di seguito.

Operatore	**Sintassi**
SHL, SHR	[Op1 : = Operatore (Op2,i)]
ROL, ROR	

Operandi:

Tipi	**Operando 1 (Op1)**	**Operando 2 (Op2)**
Parole	%MWi, %QWi, %QWAi, %SWi	%MWi, %KWi, %IW, %IWAi, %QW, %QWAi, %SWi, %BLK.x
Parole doppie	%MDi	%MDi, %KDi

Istruzioni di conversione

Introduzione

Le istruzioni di conversione eseguono la conversione tra rappresentazioni diverse di numeri.
La tabella che segue elenca i tipi di istruzioni di conversione.

Istruzione	Funzione
BTI	BCD --> Conversione binaria
ITB	Binario --> Conversione BCD

Analisi del codice BCD

BCD (Binary Coded Decimal) rappresenta una cifra decimale (da 0 a 9) mediante codifica a quattro bit. Un oggetto parola da 16 bit può quindi contenere un numero espresso da quattro cifre (0000 - 9999), e un oggetto parola doppia da 32 bit un numero espresso da otto cifre.
Durante la conversione, se il valore non è BCD, il bit di sistema %S18 viene impostato a 1. Questo bit deve essere provato e azzerato dal programma.
Rappresentazione BCD dei numeri decimali:

Decimale	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

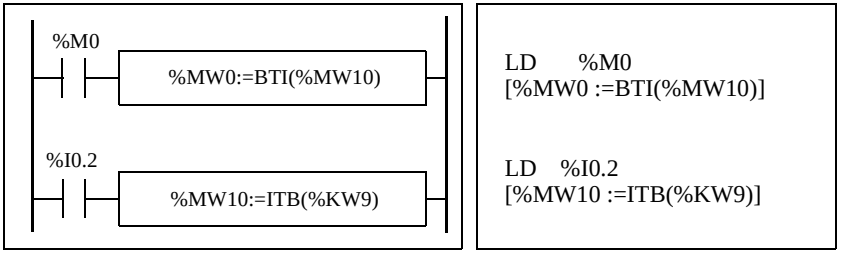
Esempi:

- La parola %MW5 esprime il valore BCD "2450" che corrisponde al valore binario: 0010 0100 0101 0000
- La parola %MW12 esprime il valore decimale "2450" che corrisponde al valore binario: 0000 1001 1001 0010

La parola %MW5 viene convertita nella parola %MW12 tramite l'istruzione BTI.
La parola %MW12 viene convertita nella parola %MW5 tramite l'istruzione ITB.

Struttura

Le operazioni di conversione vengono eseguite nel modo che segue:



Sintassi

La sintassi dipende dagli operatori utilizzati, come indicato nella tabella di seguito.

Operatore	Sintassi
BTI, ITB	[Op1 : = Operatore (Op2)]

Operandi:

Tipo	Operando 1 (Op1)	Operando 2 (Op2)
Parole	%MWi, %QWi, %QWAi, %SWi	%MWi, %KWi, %IW, %IWAi, %QW, %QWAi, %SWi, %BLK.x
Parole doppie	%MDi	%MDi, %KDi

Esempi di applicazione:

L'istruzione BTI viene utilizzata per elaborare un valore di setpoint agli ingressi del controller tramite rotelle BCD.
L'istruzione viene utilizzata per visualizzare i valori numerici (ad esempio, il risultato di un calcolo, il valore corrente di un blocco funzione) su pannelli visualizzatori codificati BCD.

Istruzioni di conversione tra parole semplici e doppie

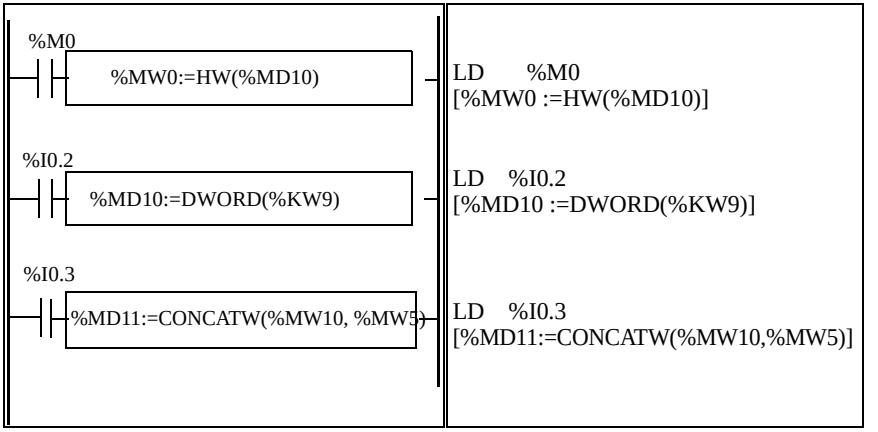
Introduzione

La tabella che segue descrive le istruzioni di conversione tra le parole semplici e doppie:

Istruzione	Funzione
LW	Estrae il bit meno significativo da una parola doppia in una parola.
HW	Estrae il bit più significativo da una parola doppia in una parola.
CONCATW	Concatena due parole per creare una parola doppia.
DWORD	Converte una parola a 16 bit in parola doppia a 32 bit.

Struttura

Le operazioni di conversione vengono eseguite nel modo che segue:



Sintassi

La sintassi dipende dagli operatori utilizzati, come indicato nella tabella di seguito: l

Operatore	Sintassi	Operando 1 (Op1)	Operando 2 (Op2)	Operando 3 (Op3)
LW, HW	Op1 = Operatore (Op2)	%MWi	%MDi, %KDi	[-]
CONCATW	Op1 = Operatore (Op2,Op3))	%MDi	%MWi, %KWi, valore immediato	%MWi, %KWi, valore immediato
DWORD	Op1 = Operatore (Op2)	%MDi	%MWi, %KWi	[-]

16.4 Istruzioni di programma

In breve

Oggetto di questa sezione

Questa sezione fornisce un'introduzione alle istruzioni di programma.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Istruzioni END	435
Istruzione NOP	437
Istruzioni di salto	438
Istruzioni di subroutine	439

Istruzioni END

Introduzione

Le istruzioni END definiscono la fine dell'esecuzione di un ciclo di programma.

END, ENDC ed ENDCN

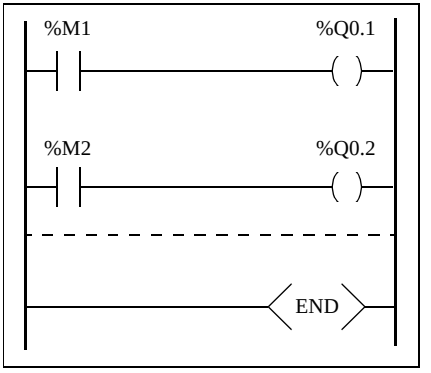
Sono disponibili tre diverse istruzioni END:

- END: fine non condizionale del programma
- ENDC: fine del programma se il risultato booleano dell'istruzione di test precedente è 1
- ENDCN: fine del programma se il risultato booleano dell'istruzione di test precedente è 0

Per impostazione predefinita (modalità normale) quando si attiva la fine del programma, le uscite vengono aggiornate e viene avviato il ciclo successivo. Se la scansione è periodica, quando si raggiunge la fine del periodo, le uscite vengono aggiornate e viene avviato il ciclo successivo.

Esempi

Esempio di un'istruzione END non condizionale.

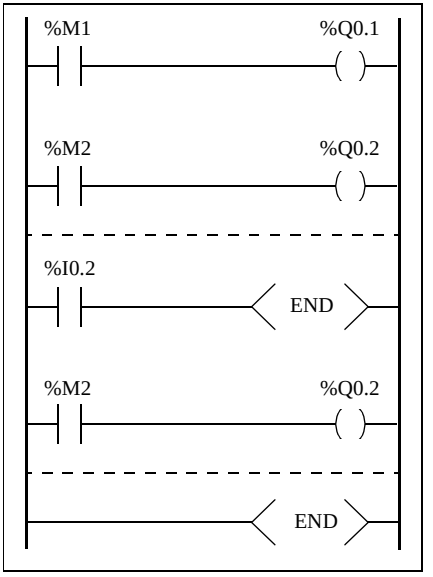



```
LD      %M1
ST      %Q0.1
LD      %M2
ST      %Q0.2

.....

END
```

Esempio di un'istruzione END condizionale.




```
LD      %M1
ST      %Q0.1
LD      %M2
ST      %Q0.2

.....

LD      %I0.2
ENDC                  → Se %I0.2 = 1, fine della scansione del programma
LD      %M2
ST      %Q0.2

                      Se %I0.2 = 0, continua la scansione del programma fino alla nuova istruzione END
.....

END
```

Istruzione NOP

NOP

L'istruzione NOP non esegue alcuna operazione. Utilizzarla per "riservare" righe in un programma in modo da poter inserire istruzioni in un secondo momento senza modificare i numeri di riga.

Istruzioni di salto

Introduzione

Le istruzioni di salto provocano l'immediata interruzione dell'esecuzione del programma e la continuazione dalla riga successiva a quella contenente l'etichetta %Li (i = da 1 a 16 per un controller compatto e da 1 a 63 per gli altri).

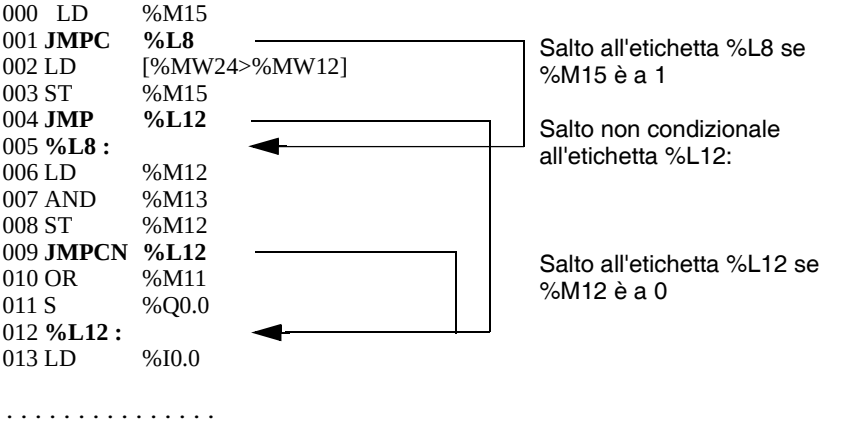
JMP, JMPC e JMPCN

Sono disponibili tre diverse istruzioni di salto:

- JMP: salto incondizionale del programma
- JMPC: salto del programma se il risultato booleano dell'operazione logica precedente è 1
- JMPCN: salto del programma se il risultato booleano dell'operazione logica precedente è 0

Esempi

Esempi di istruzioni di salto.



Indicazioni

- Le istruzioni di salto non possono essere racchiuse tra parentesi e non devono essere posizionate tra le istruzioni AND(, OR(e una parentesi chiusa ")".
- L'etichetta può essere posizionata solo prima di un'istruzione LD, LDN, LDR, LDF o BLK.
- Il numero dell'etichetta %Li deve essere definito solo una volta nel programma.
- Il salto porta a una riga precedente o successiva del programma. Se il salto è verso una riga precedente, occorre prestare attenzione al tempo di scansione del programma. Un tempo di scansione esteso può provocare l'attivazione del watchdog.

Istruzioni di subroutine

Introduzione

Le istruzioni di subroutine provocano l'esecuzione di una subroutine, quindi il ritorno al programma principale.

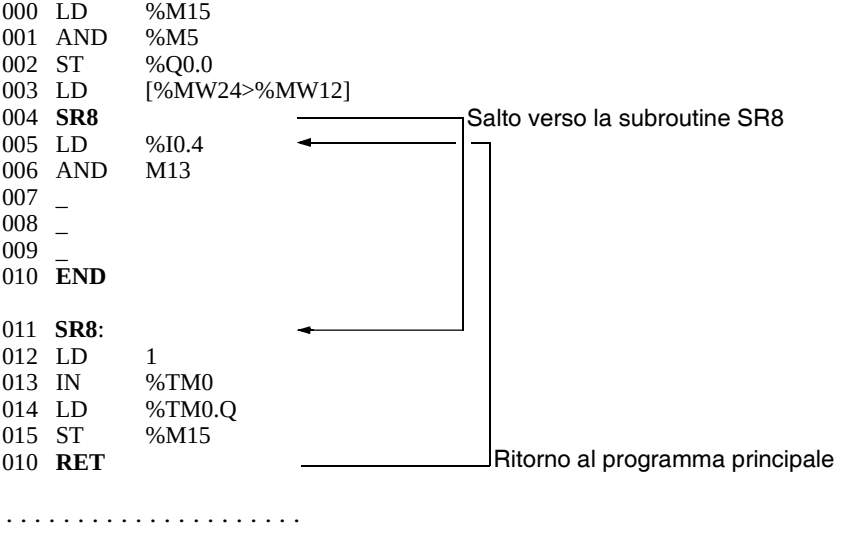
SRn, SRn: e RET

Le subroutine consistono di tre fasi:

- L'istruzione **SRn** richiama la subroutine definita dall'etichetta SRn, se il risultato della precedente istruzione booleana è 1.
- La subroutine ha come riferimento l'etichetta **SRn:**, n può assumere un valore compreso tra 0 e 15 per TWDLCAA10DRF, TWDLCAA16DRF e tra 0 e 63 per tutti gli altri controller.
- L'istruzione **RET** inserita al termine della subroutine consente il ritorno al programma principale.

Esempio

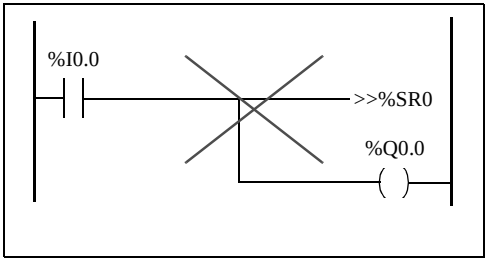
Esempi di istruzioni di subroutine.

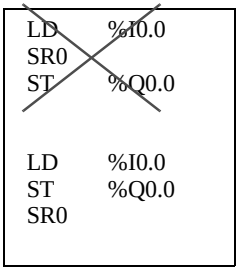


Indicazioni

- Una subroutine non deve richiamare un'altra subroutine.
- Le istruzioni di subroutine non possono essere racchiuse tra parentesi e non devono essere posizionate tra le istruzioni AND(, OR(, e un'istruzione di parentesi chiusa ")".
- L'etichetta può essere posta solo prima di un'istruzione LD o BLK che segna l'inizio di un'equazione booleana (o rete).
- La chiamata alla subroutine non deve essere seguita da un'istruzione di assegnazione. Questo perché la subroutine può cambiare il contenuto dell'accumulatore booleano. Pertanto, su un ritorno, essa può avere un valore diverso da quello posseduto prima della chiamata. Vedere l'esempio seguente.

Esempio di programmazione di una subroutine.





Istruzioni avanzate

17

In breve

Oggetto di questo capitolo

Questo capitolo fornisce informazioni sulle istruzioni e i blocchi funzione utilizzati per creare programmi di controllo avanzati per i controller Twido.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
17.1	Blocchi di funzione avanzati	443
17.2	Funzioni dell'orologio	483
17.3	Guida all'avvio rapido della funzione PID Twido	493
17.4	Funzione PID	518
17.5	Istruzioni su virgola mobile	572
17.6	Istruzioni sulle tabelle di oggetti	583

17.1 Blocchi di funzione avanzati

In breve

Oggetto di questa sezione

Questa sezione presenta i blocchi funzione avanzati, inclusi esempi di programmazione.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Oggetti bit e parola associati ai blocchi funzione avanzati	444
Principi di programmazione dei blocchi funzione avanzati	446
Blocco funzione registro LIFO/FIFO (%Ri)	448
Operazione LIFO	449
FIFO, funzionamento	450
Programmazione e configurazione dei registri	451
Blocco funzione Modulazione d'ampiezza d'impulso (%PWM)	454
Blocco funzione uscita generatore d'impulsi (%PLS)	457
Blocco funzione del Controller a tamburo (%DR)	460
Operazione %DRi del blocco funzione Controller a tamburo	461
Programmazione e configurazione dei programmatori a tamburo	463
Blocco funzione contatore veloce (%FC)	465
Blocco funzione contatore molto veloce (%VFC)	468
Trasmissione e ricezione di messaggi – Istruzione di scambio (EXCH)	479
Blocco funzione controllo di scambio (%MSGx)	480

Oggetti bit e parola associati ai blocchi funzione avanzati

Introduzione

I blocchi funzione avanzati utilizzano parole e bit dedicati dello stesso tipo dei blocchi funzione di base. I blocchi funzione avanzati includono:

- Registri LIFO/FIFO (%R)
- Programmatori a tamburo (%DR)
- Contatori veloci (%FC)
- Contatori molto veloci (%VFC)
- Uscita a modulazione d'ampiezza d'impulso (%PWM)
- Uscita generatore d'impulsi (%PLS)
- Registro bit di scorrimento (%SBR)
- Contatore passo passo (%SC)
- Blocco controllo messaggio (%MSG)

Oggetti accessibili dal programma

La tabella seguente offre una panoramica delle parole e dei bit accessibili dal programma che sono associati ai vari blocchi funzione avanzati. Si noti che l'accesso in scrittura nella tabella seguente dipende dall'impostazione "Regolabile" selezionata durante la configurazione. Questa impostazione consente o nega l'accesso alle parole o ai bit da parte di TwidoSoft o dell'interfaccia operatore.

Blocco funzione avanzato	Parole e bit associati		Indirizzo	Accesso in scrittura
%R	Parola	Ingresso del registro	%Ri.I	Sì
	Parola	Uscita del registro	%Ri.O	Sì
	Bit	Uscita registro completa	%Ri.F	No
	Bit	Uscita registro vuota	%Ri.E	No
%DR	Parola	Numero passo corrente	%DRi.S	Sì
	Bit	Ultimo passo uguale al passo corrente	%DRi.F	No
%FC	Parola	Valore corrente	%FCi.V	Sì
	Parola	Valore preimpostato	%FCi.P	Sì
	Bit	Completato	%FCi.D	No

Blocco funzione avanzato	Parole e bit associati		Indirizzo	Accesso in scrittura
%VFC	Parola	Valore corrente	%VFCi.V	No
	Parola	Valore preimpostato	%VFCi.P	Sì
	Bit	Direzione conteggio	%VFCi.U	No
	Parola	Valore di cattura	%VFCi.C	No
	Parola	Valore soglia 0	%VFCi.S0	Sì
	Parola	Valore soglia 1	%VFCi.S1	Sì
	Bit	Uscita	%VFCi.F	No
	Bit	Uscita riflessa 0 abilitata	%VFCi.R	Sì
	Bit	Uscita riflessa 1 abilitata	%VFCi.S	Sì
	Bit	Uscita soglia 0	%VFCi.TH0	No
	Bit	Uscita soglia 1	%VFCi.TH1	No
	Bit	Base tempo della misura di frequenza	%VFCi.T	Sì
%PWM	Parola	Percentuale di impulso al passo 1 in relazione al periodo totale.	%PWMi.R	Sì
	Parola	Periodo preimpostato	%PWMi.P	Sì
%PLS	Parola	Numero di impulsi	%PLSi.N	Sì
	Parola	Valore preimpostato	%PLSi.P	Sì
	Bit	Uscita corrente abilitata	%PLSi.Q	No
	Bit	Generazione terminata	%PLSi.D	No
%SBR	Bit	Bit di registro	%SBRi.J	No
%SC	Bit	Bit contatore passo	%SCi.j	Sì
%MSG	Bit	Completato	%MSGi.D	No
	Bit	Errore	%MSGi.E	No

Principi di programmazione dei blocchi funzione avanzati

In breve

Le applicazioni Twido sono memorizzate sotto forma di programmi List, anche se sono state scritte con un editor Ladder. I controller Twido possono quindi essere considerati come "macchine List". Il termine "reversibilità" si riferisce alla capacità di TwidoSoft di rappresentare un'applicazione List come Ladder e viceversa. Per default, tutti i programmi Ladder sono reversibili.
Come nel caso dei blocchi funzione di base, i blocchi funzione avanzati devono prendere in considerazione anche le regole di reversibilità. La struttura dei blocchi funzione reversibili nel linguaggio List richiede l'utilizzo delle seguenti istruzioni:

- **BLK**: Contrassegna l'inizio del blocco e la parte di ingresso del blocco funzione
- **OUT_BLK** : Contrassegna l'inizio della parte di uscita del blocco funzione
- **END_BLK** : Contrassegna la fine del blocco funzione

Nota: L'utilizzo delle istruzioni di blocco funzione reversibile non è obbligatorio in un programma List correttamente funzionante. Per alcune istruzioni è possibile programmare in List, linguaggio non reversibile.

Ingressi e uscite dedicate

Le funzioni avanzate Contatore veloce (FC), Contatore molto veloce (VFC), PLS e PWM utilizzano ingressi e uscite dedicati. Questi bit non sono tuttavia riservati all'uso esclusivo da parte di un solo blocco. L'utilizzo di queste risorse dedicate deve essere gestito.
Quando si utilizzano queste funzioni avanzate, è necessario gestire il modo in cui vengono assegnati gli ingressi e le uscite dedicati. TwidoSoft aiuta nella configurazione di queste risorse visualizzando i dettagli di configurazione di ingressi/uscite ed emette un avvertimento se un ingresso o un'uscita dedicati è già utilizzato da un blocco funzione configurato.
Le tabelle che seguono riassumono le dipendenze di ingressi e uscite dedicati e delle funzioni specifiche.
Se utilizzati nelle funzioni di conteggio:

Ingressi	Utilizzo
%I0.0.0	%VFC0: Gestione avanti/indietro o fase B
%I0.0.1	%VFC0: Ingresso a impulsi o fase A
%I0.0.2	%FC0: Ingresso a impulsi o ingresso preimpostato %VFC0
%I0.0.3	%FC1: Ingresso a impulsi o ingresso di cattura %VFC0
%I0.0.4	%FC2: Ingresso a impulsi o ingresso di cattura %VFC1
%I0.0.5	Ingresso preimpostato %VFC1
%I0.0.6	%VFC1: Gestione avanti/indietro o fase B
%I0.0.7	%VFC1: Ingresso a impulsi o fase A

Se utilizzati nelle funzioni di conteggio o speciali:

Uscite	Utilizzo
%Q0.0.0	%PLS0 o uscita PWM0
%Q0.0.1	%PLS1 o uscita PWM1
%Q0.0.2	Uscite di riflesso per %VFC0
%Q0.0.3	
%Q0.0.4	Uscite di riflesso per VFC1
%Q0.0.5	

Ingressi e uscite dedicati

TwidoSoft applica le seguenti regole per l'utilizzo di ingressi e uscite dedicati.

- Ciascun blocco funzione che utilizza I/O dedicati deve essere configurato e quindi avere un riferimento nell'applicazione. Gli I/O sono assegnati solo durante la configurazione di un blocco funzione. Non vengono quindi assegnati durante l'uso in un programma.
- Dopo aver configurato un blocco funzione, i relativi ingressi e uscite dedicati non possono essere utilizzati dall'applicazione o da un altro blocco funzione.
  Ad esempio, se si configura %PLS0, non è possibile utilizzare %Q0.0.0 in %DR0 (controller a tamburo) o nella logica dell'applicazione (ossia, ST %Q0.0.0).
- Se un blocco funzione richiede un ingresso o un'uscita dedicati già utilizzati dall'applicazione o da un altro blocco funzione, questo blocco funzione non può essere configurato.
  Ad esempio, se si configura %FC0 come contatore avanti, non è possibile configurare %VFC0 in modo da utilizzare %I0.0.2 come ingresso di cattura.

Nota: Se si desidera modificare l'utilizzo di I/O dedicati, prima è necessario annullare la configurazione del blocco funzione impostando il tipo di oggetto a "non usato", quindi rimuovere il riferimento al blocco funzione nell'applicazione.

Blocco funzione registro LIFO/FIFO (%Ri)

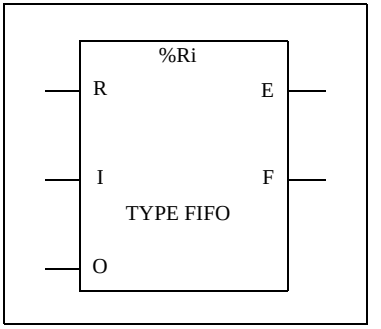
Introduzione

Un registro è un blocco di memoria in grado di memorizzare fino a 16 parole di 16 bit ciascuna in due modi diversi:

- coda FIFO (First In, First Out);
- Stack LIFO (Last In, First Out)

Illustrazione

Di seguito viene illustrato il blocco funzione Registro.



Blocco funzione Registro

Impostazioni

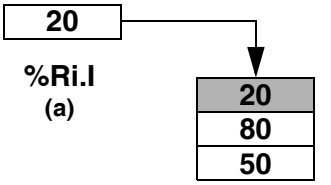
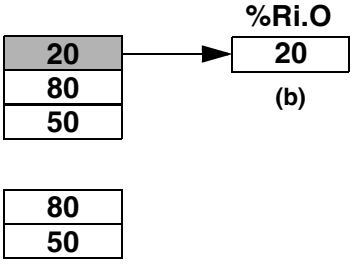
Il blocco funzione Registro presenta i seguenti parametri:

Parametro	Etichetta	Valore
Numero registro	%Ri	da 0 a 3
Tipo	FIFO o LIFO	Coda o Stack)
Parola d'ingresso	%Ri.I	Parola d'ingresso del registro. La parola può essere letta, provata e scritta.
Parola d'uscita	%Ri.O	Parola d'uscita del registro. La parola può essere letta, provata e scritta.
Ingresso di memorizzazione (o istruzione)	I (In)	Su un fronte di salita, il contenuto della parola %Ri.I nel registro.
Ricerca ingresso (o istruzione)	O (Output, Uscita)	Su un fronte di salita, carica una parola dati del registro nella parola %Ri.O.
Ingresso (o istruzione) Reset	R (Reset)	Nello stato 1, inizializza il registro.
Uscita vuota	E (Empty, Vuoto)	Il bit associato %Ri.E indica che il registro è vuoto. È possibile testarlo.
Uscita completa	F (Full)	Il bit associato %Ri.F indica che il registro è completo. È possibile testarlo.

Operazione LIFO

Introduzione Nell'operazione LIFO (Last In, First Out), l'ultima voce dati immessa è la prima ad essere trovata.

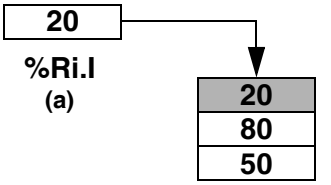
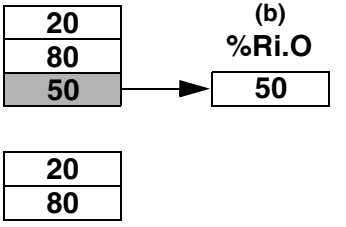
Funzionamento La tabella seguente descrive l'operazione LIFO.

Passo	Descrizione	Esempio
1	Quando viene ricevuta una richiesta di memorizzazione (fronte di salita all'ingresso I o attivazione dell'istruzione I), il contenuto della parola d'ingresso %Ri.I (già caricata) viene memorizzato nella parte superiore dello stack (Fig. a). Se lo stack è pieno (uscita F=1), non è più possibile alcuna memorizzazione.	Memorizzazione del contenuto di %Ri.I nella parte superiore dello stack. 
2	Quando viene ricevuta una richiesta di ricerca (fronte di salita all'ingresso O o attivazione dell'istruzione O), la parola dati più alta (l'ultima parola immessa) viene caricata nella parola %Ri.0 (Fig. b). Se il registro è vuoto (uscita E=1) non è più possibile alcuna ricerca. La parola d'uscita %Ri.O non cambia e mantiene l'ultimo valore assunto.	Ricerca della parola dati più alta nello stack. 
3	Lo stack può essere azzerato in qualsiasi momento (stato 1 all'ingresso R o attivazione dell'istruzione R). L'elemento indicato dal puntatore è quindi il più alto nello stack.	

FIFO, funzionamento

Introduzione Nell'operazione FIFO (First In, First Out), la prima voce dati immessa è la prima ad essere trovata.

Funzionamento La tabella seguente descrive l'operazione FIFO.

Passo	Descrizione	Esempio
1	Quando viene ricevuta una richiesta di memorizzazione (fronte di salita all'ingresso I o attivazione dell'istruzione I), il contenuto della parola d'ingresso %Ri.I (già caricata) viene memorizzato nella parte superiore della coda (Fig. a). Se la coda è piena (uscita F=1), non è più possibile alcuna memorizzazione.	Memorizzazione del contenuto di %Ri.I nella parte superiore della coda. 
2	Quando viene ricevuta una richiesta di ricerca (fronte di salita all'ingresso O o attivazione dell'istruzione O), la parola dati più bassa nella coda viene caricata nella parola d'uscita %Ri.O e il contenuto del registro viene spostato in basso di una posizione nella coda (Fig. b). Se il registro è vuoto (uscita E=1) non è più possibile alcuna ricerca. La parola d'uscita %Ri.O non cambia e mantiene l'ultimo valore assunto.	Ricerca della prima voce dati caricata in %Ri.O. 
3	La coda può essere azzerata in qualsiasi momento (stato 1 all'ingresso R o attivazione dell'istruzione R).	

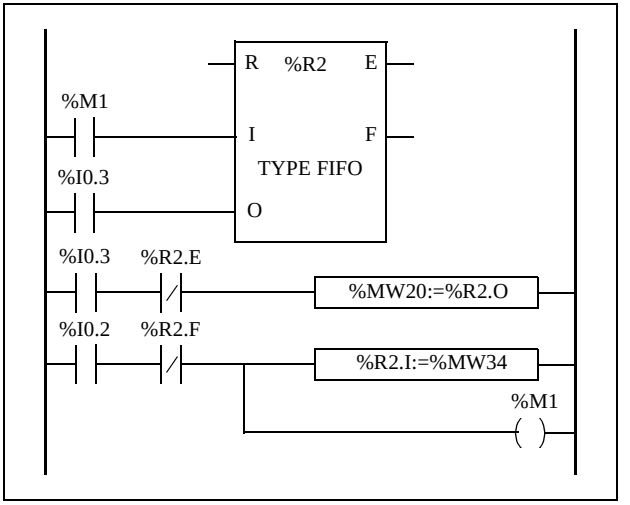
Programmazione e configurazione dei registri

Introduzione

Il seguente esempio di programmazione mostra il contenuto di una parola di memoria (%MW34) caricata in un registro (%R2.I) alla richiesta di memorizzazione (%I0.2), se il registro %R2 non è pieno (%R2.F = 0). La richiesta di memorizzazione nel registro è eseguita da %M1. La richiesta di ricerca è eseguita dall'ingresso %I0.3 e %R2.O viene caricata in %MW20, se il registro non è vuoto (%R2.E = 0).

Esempio di programmazione

L'illustrazione che segue mostra un blocco funzione Registro con esempi di programmazione reversibile e non reversibile.



Ladder

```
BLK          %R2
LD           %M1
I
LD           %I0.3
O
END_BLK
LD           %I0.3
ANDN         %R2.E
[%MW20:=%R2.O]
LD           %I0.2
ANDN         %R2.F
[%R2.I:=%MW34]
ST           %M1
```

Programma reversibile

```
LD           %M1
I            %R2
LD           %I0.3
O            %R2
ANDN         %R2.E
[%MW20:=%R2.O]
LD           %I0.2
ANDN         %R2.F
[%R2.I:=%MW34]
ST           %M1
```

Programma non reversibile

Configurazione

L'unico parametro che deve essere immesso durante la configurazione è il tipo di registro:

- FIFO (default) oppure
- LIFO

Casi speciali

La tabella seguente contiene un elenco di casi speciali di funzionamento.

Caso speciale	Descrizione
Effetto di un riavvio a freddo (%S0=1)	Inizializza il contenuto del registro. Il bit d'uscita %Ri.E associato all'uscita E è impostato a 1.
Effetto di un riavvio a caldo (%S1=1) di un arresto del controller	Non ha effetto sul valore corrente del registro, né sullo stato dei relativi bit d'uscita.

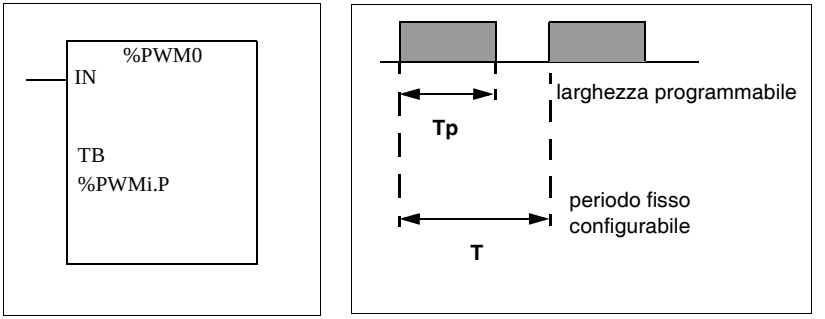
Blocco funzione Modulazione d'ampiezza d'impulso (%PWM)

Introduzione

Il blocco funzione Modulazione d'ampiezza d'impulso (%PWM) genera un segnale a onde rettangolari sui canali d'uscita dedicati (%Q0.0.0 o %Q0.0.1), di cui è possibile variare l'ampiezza e, quindi, il rapporto ciclico. I controller con uscite a relè per questi due canali non supportano tale funzione a causa della limitazione della frequenza.
Sono disponibili due blocchi %PWM. %PWM0 utilizza l'uscita dedicata %Q0.0.0 e %PMW1 utilizza l'uscita dedicata %Q0.0.1. I blocchi funzione %PLS condividono le stesse uscite dedicate. È quindi necessario scegliere una o l'altra funzione.

Illustrazione

Blocco PWM e diagramma dei tempi



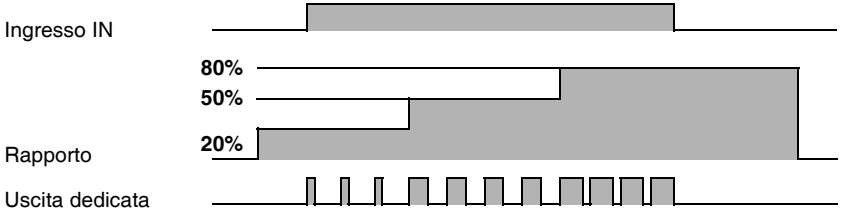
Impostazioni La tabella seguente elenca i parametri del blocco funzione PWM.

Parametro	Etichetta	Descrizione
Base tempo	TB	0,142 ms, 0,57 ms, 10 ms, 1 s (valore predefinito)
Preselezione del periodo	%PWMi.P	0 < %PWMi.P <= 32.767 con base tempo di 10 ms o 1 s 0 < %PWMi.P <= 255 con base tempo di 0,57 ms o 0,142 ms 0 = Funzione non utilizzata
Rapporto ciclico	%PWMi.R	Il valore fornisce la percentuale del segnale nello stato 1 in un periodo. L'ampiezza Tp è pertanto uguale a: Tp = T * (%PWMi.R/100). L'applicazione utente scrive il valore di %PWMi.R. Questa parola controlla il rapporto ciclico del periodo. Per la definizione T, vedere "Intervallo di periodi", di seguito. Il valore predefinito è 0 e i valori maggiori di 100 sono considerati uguali a 100.
Ingresso di generazione d'impulsi	IN	Allo stato 1, il segnale di modulazione d'ampiezza d'impulso viene generato sul canale d'uscita. Allo stato 0, il canale d'uscita è impostato a 0.

Intervallo di periodi Il valore preimpostato e la base tempo possono essere modificati durante la configurazione. Tali parametri sono utilizzati per fissare il periodo del segnale T=%PWMi.P * TB. Minore è il rapporto da ottenere, maggiore deve essere il %PWMi.P selezionato. Intervallo di periodi disponibile:

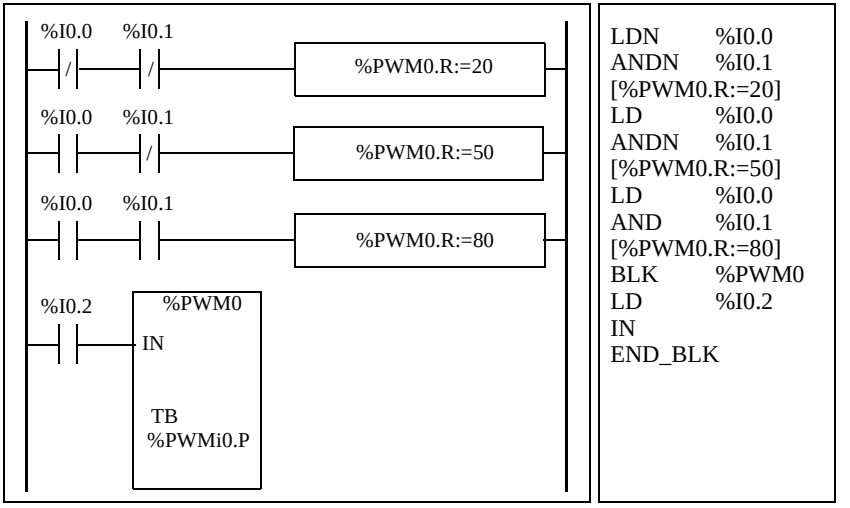
- da 0,142 ms a 36,5 ms in passi di 0,142 ms (da 27,4 Hz a 7 kHz)
- da 0,57 ms a 146 ms in passi di 0,57 ms (da 6,84 Hz a 1,75 kHz)
- da 10 ms a 5,45 min in passi da 10 ms
- da 1 s a 9,1 ore in passi di 1 s

Funzionamento La frequenza del segnale d'uscita è impostata durante la configurazione selezionando la base tempo TB e il %PWMi.P preimpostato. La modifica del rapporto ciclico % PWMi.R nel programma modula l'ampiezza del segnale. Di seguito è riportata un'illustrazione del diagramma d'impulsi del blocco funzione PWM con rapporti ciclici diversi.



Programmazione e configurazione

In questo esempio, l'ampiezza del segnale viene modificata dal programma in base allo stato degli ingressi del controller %I0.0.0 e %I0.0.1.
Se %I0.1 e % I0.0.2 sono impostati a 0, il rapporto %PWM0.R è impostato a 20%, la durata del segnale allo stato 1 è quindi: 20 % x 500 ms = 100 ms.
Se %I0.0.0 è impostato a 0 e %I0.0.1 a 1, il rapporto %PWM0.R è impostato al 50 % (durata di 250 ms).
Se %I0.0.0 e %I0.0.1 sono impostati a 1, il rapporto %PWM0.R è impostato all'80% (durata 400 ms).
Esempio di programmazione:



Casi speciali

La tabella seguente contiene un elenco di casi speciali per il funzionamento del blocco funzione PWM.

Caso speciale	Descrizione
Effetto di un riavvio a freddo (%S0=1)	Imposta il rapporto %PWMi.R a 0. Inoltre, il valore di %PWMi.P è reimpostato al valore configurato ed è prioritario rispetto a qualsiasi modifica apportata con l'editor della tabella delle animazioni o il pannello visualizzatore opzionale.
Effetto di un riavvio a caldo (%S1=1)	Non ha alcun effetto.
Conseguenza del fatto che le uscite sono dedicate al blocco %PWM	Forzando l'uscita %Q0.0.0 o %Q0.0.1 utilizzando una periferica di programmazione non viene interrotta la generazione del segnale.

Blocco funzione uscita generatore d'impulsi (%PLS)

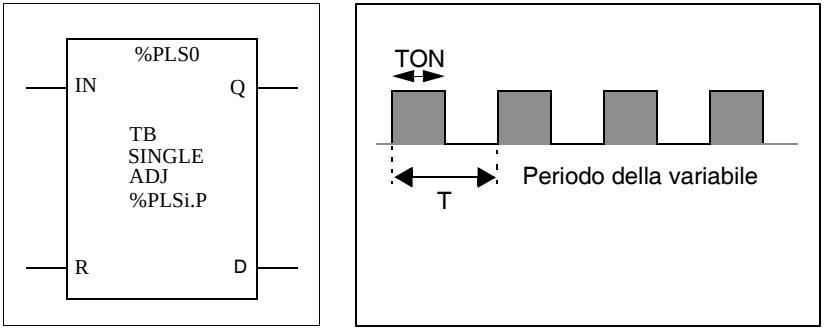
Introduzione

Il blocco funzione %PLS è utilizzato per generare segnali a onda quadra. Sui canali d'uscita dedicati %Q0.0.0 e %Q0.0.1 sono disponibili due funzioni %PLS. Il blocco funzione %PLS consente una sola ampiezza del segnale, o ciclo duty, del 50%. È possibile scegliere di limitare il numero di impulsi oppure il periodo quando viene eseguita la sequenza di impulsi. Questi valori possono essere determinati al momento della configurazione oppure aggiornati dall'applicazione utente.

Nota: i controller con uscite a relè per questi due canali non supportano la funzione %PLS.

Rappresentazione

Esempio del blocco funzione del generatore di impulsi in modalità parola singola:



- TON=T/2 per le basi tempo 0,142 ms e 0,57 ms
  = (%PLSi.P*TB)/2
- TON=[parte intera(%PLSi.P)/2]*TB per le basi tempo da 10 ms a 1s.

Specifiche tecniche

La tabella seguente contiene le caratteristiche del blocco funzione PLS:

Funzione	Oggetto	Descrizione
Base tempo	TB	0,142 ms, 0,57 ms, 10 ms, 1 s
Periodo preimpostato	%PLSi.P	Gli impulsi sull'uscita %PLS1 non si interrompono quando viene raggiunto %PLS1.N o %PLS1.ND* per basi tempo di 0,142 ms e 0,57 ms. ● 1 < %PLSi.P <= 32767 con una base tempo di 10 ms o 1 s ● 0 < %PLSi.P <= 255 con una base tempo di 0,57 ms o 0,142 ms ● 0 = Funzione non utilizzata. Per ottenere una buona precisione del rapporto ciclico con le basi tempo di 10 ms e 1s, è opportuno che %PLSi >= 100 se P è dispari.
Numero di impulsi	%PLSi.N %PLSi.ND*	Il numero di impulsi da generare nel periodo T può essere limitato all'intervallo 0 <= %PLSi.N <= 32767 in modalità standard o 0 <= %PLSi.ND <= 4294967295 in modalità parola doppia. Il valore predefinito è impostato a 0. Per produrre un numero illimitato di impulsi, impostare %PLSi.N o %PLSi.ND a zero. Il numero di impulsi può essere sempre modificato indipendentemente dall'impostazione dell'opzione Regolabile.
Regolabile	Y/N	Se impostato a Y, è possibile modificare il valore preimpostato %PLSi.P tramite pannello visualizzatore o l'editor delle tabelle di animazione. Se impostato a N, non c'è accesso alla preimpostazione.
Ingresso di generazione d'impulsi	IN	Allo stato 1, la generazione d'impulsi viene prodotta sul canale d'uscita dedicato. Allo stato 0, il canale d'uscita è impostato a 0.
Azzera ingresso	R	Allo stato 1, le uscite %PLSi.Q e %PLSi.D sono impostate a 0. Il numero di impulsi generato nel periodo T è impostato a 0.
Generazione uscita impulso corrente	%PLSi.Q	Allo stato 1, indica che il segnale a impulsi viene generato sul canale d'uscita dedicato configurato.
Uscita di conferma della generazione d'impulsi	%PLSi.D	Allo stato 1 la generazione del segnale è completa. È possibile raggiungere il numero di impulsi desiderato.

Nota: (*) significa una variabile parola doppia.

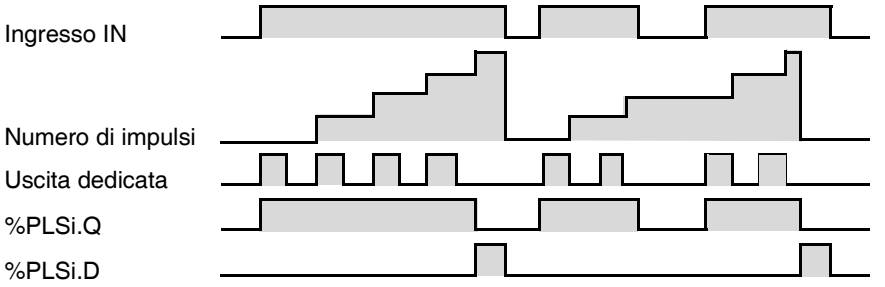
Intervallo di periodi

Il valore preimpostato e la base tempo possono essere modificati durante la configurazione. Questi parametri sono utilizzati per fissare il periodo del segnale T=%PLSi.P * TB. Intervallo di periodi disponibile:

- da 0,142 ms a 36,5 ms in passi di 0,142 ms (da 27,4 Hz a 7 kHz)
- da 0,57 ms a 146 ms in passi da 0,57 ms (da 6,84 Hz a 1,75 kHz)
- da 20 ms a 5,45 min in passi da 10 ms
- da 2 s a 9,1 ore in passi da 1 s

Funzionamento

Di seguito viene riportata l'illustrazione del blocco funzione %PLS.



Casi speciali

Caso speciale	Descrizione
Effetto di un avvio a freddo (%S0=1)	Imposta %PLSi.P al valore definito durante la configurazione
Effetto di un riavvio a caldo (%S1=1)	Non ha alcun effetto.
Effetto della modifica della preimpostazione (%PLSi.P)	È immediatamente attivo
Conseguenza del fatto le uscite sono dedicate al blocco %PLS	Forzando l'uscita %Q0.0.0 o %Q0.0.1 utilizzando una periferica di programmazione non viene interrotta la generazione del segnale.

Nota: %PLSx.D viene definito dopo la generazione del numero di impulsi desiderato. Viene reimpostato regolando gli ingressi IN o R su 1.

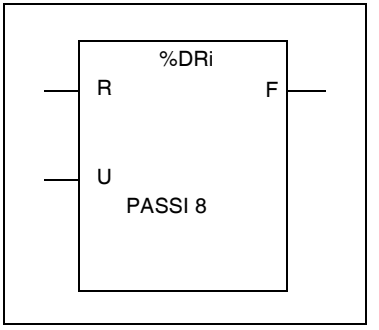
Blocco funzione del Controller a tamburo (%DR)

Introduzione

Il controller a tamburo funziona in base a un principio simile a quello di un tamburo elettromeccanico che cambia passo in base agli eventi esterni. In ciascun passo, il punto alto di una camma emette un comando che viene eseguito dall'automazione. Nel caso di un controller a tamburo, questi punti alti sono simboleggiati dallo stato 1 per ciascun passo e sono assegnati ai bit di uscita %Qi.j o ai bit interni %Mi, noti come bit di controllo.

Illustrazione

Di seguito viene illustrato il blocco funzione Controller a tamburo.



Blocco funzione Controller a tamburo

Impostazioni

Il blocco funzione del controller a tamburo presenta i seguenti parametri:

Parametro	Etichetta	Valore
Numero	%DRi	da 0 a 3 controller compatto da 0 a 7 controller modulari
Numero passo corrente	%DRi.S	0<%DRi.S<7. Parola che può essere letta e scritta. Il valore scritto deve essere un valore immediate decimale. Quando viene scritto, il nuovo valore è effettivo alla successiva esecuzione del blocco funzione.
Numero di passi		da 1 a 8 (predefiniti)
Ingresso ritorno al passo 0 (o all'istruzione)	R (Reset)	Allo stato 1, imposta il controller a tamburo sul passo 0.
Ingresso avanzamento (o istruzione)	U (alto)	Su un fronte di salita, fa avanzare di un passo il controller a tamburo e aggiorna i bit di controllo.
Uscita	F (Full)	Indica che il passo corrente è uguale all'ultimo passo definito. Il bit %DRi.F associato può essere provato (ad esempio, %DRi.F=1, se %DRi.S = numero di passi configurati - 1).
Bit di controllo		Le uscite o i bit interni associati al passo (bit di controllo 16) e definiti nell'editor di configurazione.

Operazione %DRi del blocco funzione Controller a tamburo

Introduzione

Il controller a tamburo è composto da:

- Una matrice di dati costanti (le camme) strutturata in otto passi (da 0 a 7) e 16 bit di dati (stato del passo) disposti in colonne numerate da 0 a F.
- Un elenco di bit di controllo è associato a un'uscita configurata (%Qi.j.k) o a una parola di memoria (%Mi). Durante il passo corrente, i bit di controllo assumono gli stati binari definiti per questo passo.

L'esempio della tabella seguente riepiloga le caratteristiche principali del controller a tamburo.

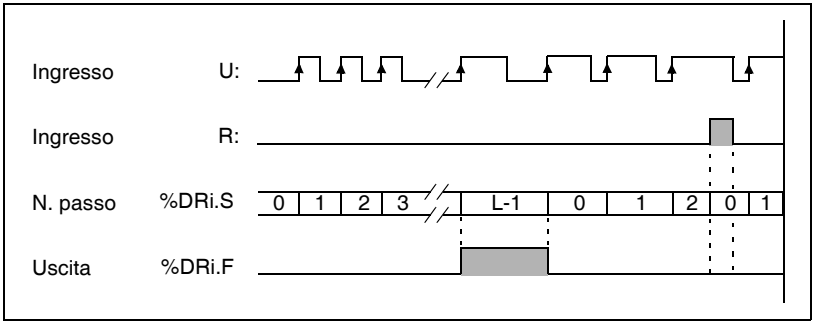
Colonna	**0**	**1**	**2**		**D**	**E**	**F**
Bit di controllo	%Q0.1	%Q0.3	%Q1.5		%Q0.6	%Q0.5	%Q1.0
Passo 0	0	0	1		1	1	0
Passo 1	1	0	1		1	0	0
Passo 5	1	1	1		0	0	0
Passo 6	0	1	1		0	1	0
Passo 7	1	1	1		1	0	0

Funzionamento

Nell'esempio precedente, il passo 5 è il passo corrente, i bit di controllo %Q0.1, %Q0.3 e %Q1.5 sono allo stato 1, i bit di controllo %Q0.6, %Q0.5 e %Q1.0 sono allo stato 0. Il numero di passo corrente viene incrementato di un'unità su ogni fronte di salita dell'ingresso U (o all'attivazione dell'istruzione U). Il passo corrente può essere modificato dal programma.

Diagramma dei tempi

Il diagramma seguente mostra il funzionamento del controller a tamburo.



Casi speciali

La tabella seguente contiene un elenco di casi speciali per il funzionamento del controller a tamburo.

Caso speciale	Descrizione
Effetti di un riavvio a freddo (%S0=1)	Reimposta il controller a tamburo al passo 0 (aggiornando i bit di controllo).
Effetto di un riavvio a caldo (%S1=1)	Aggiorna i bit di controllo dopo il passo corrente.
Effetto di un salto di programma	Il fatto che non venga più eseguita la scansione del controller a tamburo non riporta a zero i bit di controllo.
Aggiornamento dei bit di controllo	Si verifica solo quando interviene un cambiamento di passo o in caso di riavvio a caldo o a freddo.

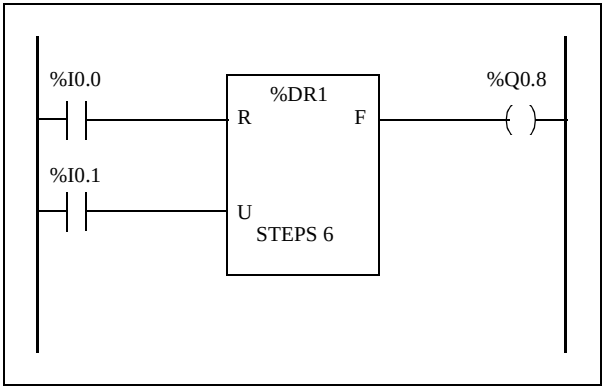
Programmazione e configurazione dei programmatori a tamburo

Introduzione

L'esempio seguente mostra la programmazione e la configurazione di un programmatore a tamburo, dove le prime sei uscite da %Q0.0 a %Q0.5 vengono abilitate in successione ogni volta che l'ingresso %I0.1 è impostato a 1. L'ingresso I0.0 reimposta le uscite a 0.

Esempio di programmazione

L'illustrazione che segue mostra un blocco funzione programmatore a tamburo con esempi di programmazione reversibile e non reversibile.



Schema Ladder

```
BLK     %DR1
LD      %I0.0
R
LD      %I0.1
U
OUT_BLK
LD      F
ST      %Q0.8
END_BLK
```

Configurazione

Durante la configurazione vengono definite le seguenti informazioni:

- numero di passi: 6
- gli stati delle uscite (bit di controllo) per ciascun passo del programmatore a tamburo.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Fase 1:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fase 2:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fase 3:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Fase 4:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Fase 5:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Fase 6:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

- Assegnazione dei bit di controllo.

1 :	%Q0.0	4 :	%Q0.1
2 :	%Q0.2	5 :	%Q0,3
3 :	%Q0.4	6 :	%Q0.5

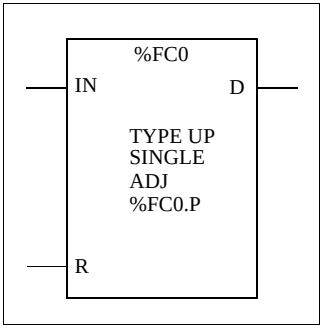
Blocco funzione contatore veloce (%FC)

Introduzione

Il blocco funzione Contatore veloce (%FC) funge da contatore avanti e indietro. È in grado di contare il fronte di salita di ingressi digitali fino alle frequenze di 5 kHz in modalità di calcolo a parola singola o a parola doppia. Dato che i contatori veloci sono gestiti da interrupt hardware specifici, il mantenimento delle velocità massime di campionatura della frequenza può variare in base all'applicazione specifica e alla configurazione hardware.
I controller compatti TWDLCA•40DRF possono supportare fino a quattro contatori veloci, mentre tutte le altre serie di controller compatti possono essere configurate per l'uso di max. tre contatori veloci. I controller modulari possono usare al massimo due contatori veloci. I blocchi funzione Contatore veloce %FC0, %FC1, %FC2 e %FC3 utilizzano rispettivamente gli ingressi dedicati %I0.0.2, %I0.0.3 e %I0.0.4. Questi bit non sono riservati per il loro uso esclusivo. La loro assegnazione deve essere determinata in base all'utilizzo di queste risorse dedicate da parte di altri blocchi funzione.

Illustrazione

Di seguito è riportato un esempio di un blocco funzione contatore veloce in modalità parola singola.



Impostazioni La tabella seguente elenca i parametri del blocco funzione Contatore veloce.

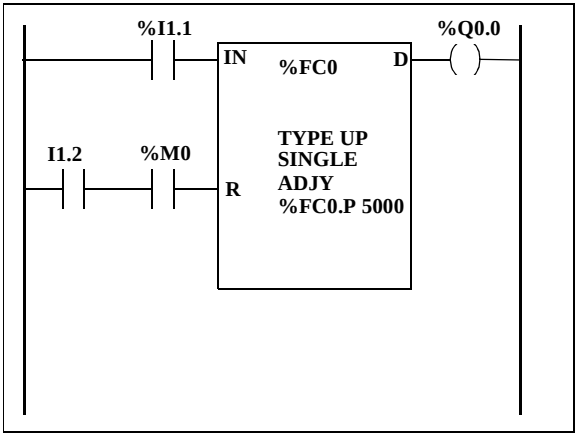
Parametro	Etichetta	Descrizione
Funzione	TYPE	Selezionato alla configurazione, può essere impostato per il conteggio in avanti o indietro.
Valore preimpostato	%FCi.P %FCi.PD	Il valore iniziale può essere impostato: ->tra 1 e 65635 in modalità standard, ->tra 1 e 4294967295 in modalità parola doppia,
Regolabile	Y/N	Se impostato su Y, è possibile modificare il valore predefinito %FCi.P e %FCi.PD e il valore corrente %FCi.V o %FCi.VD con il pannello visualizzatore o l'editor delle tabelle di animazione. Se impostato a N, non c'è accesso alla preimpostazione.
Valore corrente	%FCi.V %FCi.VD	Il valore corrente conta l'incremento o il decremento in base alla funzione di conteggio avanti o indietro selezionata. Per il conteggio in avanti, il valore di conteggio corrente viene aggiornato e può arrivare a 65535 in modalità standard (%FCi.V) e a 4294967295 in modalità parola doppia (%FCi.VD). Per il conteggio indietro, il valore corrente è il valore di preselezione %FCi.P o %FCi.PD e può diminuire fino a zero.
Ingresso di convalida	IN	Allo stato 1, il valore corrente viene aggiornato in base agli impulsi applicati all'ingresso fisico. Allo stato 0, il valore corrente resta invariato.
Ripristino	%FCi.R	Utilizzato per inizializzare il blocco. Allo stato 1, il valore corrente viene azzerato se configurato come contatore avanti o impostato a %FCi.P o %FCi.PD se configurato come contatore indietro. Il bit di conferma %FCi.D viene reimpostato al valore di default.
Completato	%FCi.D	Questo bit è impostato a 1 quando %FCi.V o %FCi.VD raggiunge il %FCi.P o %FCi.PD configurato come contatore avanti o quando %FCi.V o %FCi.VD raggiunge il valore zero quando configurato come contatore indietro. Questo bit a sola lettura viene azzerato solo impostando %FCi.R a 1.

Nota speciale Se configurato in modo da essere regolabile, l'applicazione può modificare il valore predefinito %FCi.P o %FCi.PD e il valore corrente %FCi.V o %FCi.VD in qualsiasi momento. Tuttavia, un valore nuovo viene preso in considerazione solo se è attivo l'azzeramento degli ingressi o sul fronte di salita dell'uscita %FCi.D. Questo consente diversi conteggi successivi senza la perdita di un solo impulso.

Funzionamento Se configurato per il conteggio avanti, quando un fronte di salita appare all'ingresso dedicato, il valore corrente viene aumentato di uno. Il bit di uscita %TMi.Q viene impostato a 1 quando il valore corrente raggiunge %TMi.P.
Se configurato per il conteggio all'indietro, quando un fronte di salita appare sull'ingresso dedicato, il valore corrente viene diminuito di uno. Quando il valore è zero, il bit dell'uscita di conferma %FCi.D è impostato a 1 e il valore di preselezione viene caricato nel valore corrente %FCi.V o %FCi.VD.

Configurazione e programmazione

In questo esempio, l'applicazione conta un numero di elementi fino a 5000 mentre %I1.1 è impostato a 1. L'ingresso per %FC0 è l'ingresso dedicato %I0.0.2. Quando si raggiunge il valore di preselezione, %FC0.D è impostato a 1 e conserva lo stesso valore fino a quando %FC0.R non viene comandato dal risultato dell'operazione booleana "AND" su %I1.2 e %M0.




```
BLK      %FC0
LD       %I1.1
IN
LD       %I1.2
AND      %M0
R
OUT_BLK
LD D
ST %Q0.0
END_BLK
```

Casi speciali

La tabella seguente contiene un elenco dei casi speciali di funzionamento del blocco funzione %FC.

Caso speciale	Descrizione
Effetto di un avvio a freddo (%S0=1)	Reimposta tutti gli attributi %FC con i valori configurati dall'utente o dall'applicazione utente.
Effetto di un riavvio a caldo (%S1=1)	Non ha alcun effetto.
Effetto di un arresto del controller	%FC continua a contare con le impostazioni di parametro attive al momento in cui il controller è stato interrotto.

Blocco funzione contatore molto veloce (%VFC)

Introduzione

Il blocco funzione Contatore molto veloce (%VFC) può essere configurato da TwidoSoft in modo che esegua una delle seguenti funzioni:

- Contatore avanti/indietro
- Contatore bifase avanti/indietro
- Contatore avanti singolo
- Contatore indietro singolo
- Frequenzimetro

Il blocco funzione %VFC supporta il conteggio di ingressi digitali fino alle frequenze di 20 kHz in modalità di calcolo a parola singola o a parola doppia. I controller compatti TWDLCA•40DRF possono supportare fino a due contatori molto veloci, mentre tutte le altre serie di controller compatti possono configurare un solo contatore molto veloce (%VFC). I controller modulari possono configurare fino a due contatori molto veloci (%VFC).

Assegnazioni di I/O dedicati

I blocchi funzione Contatore molto veloce (%VFC) utilizzano ingressi dedicati e ingressi e uscite ausiliari. Tali ingressi e uscite non sono riservati per l'uso esclusivo. La loro assegnazione deve essere determinata in base all'uso di queste risorse dedicate da parte di altri blocchi funzione. Le assegnazioni sono riepilogate nella tabella seguente:

		Ingressi principali		**Ingressi ausiliari**		**Uscite riflesse**	
%VFC0	Uso selezionato	Ingresso IA	Ingresso IB	IPres	Ica	Uscita 0	Uscita 1
	Contatore avanti/ indietro	%I0.0.1	%I0.0.0 (UP=0/DO=1)	%I0.0.2 (1)	%I0.0.3 (1)	%Q0.0.2 (1)	%Q0.0.3 (1)
	Contatore bifase avanti/indietro	%I0.0.1	%I0.0.0 (Impulso)	%I0.0.2 (1)	%I0.0.3 (1)	%Q0.0.2 (1)	%Q0.0.3 (1)
	Contatore avanti singolo	%I0.0.1	(2)	%I0.0.2 (1)	%I0.0.3 (1)	%Q0.0.2 (1)	%Q0.0.3 (1)
	Contatore indietro singolo	%I0.0.1	(2)	%I0.0.2 (1)	%I0.0.3 (1)	%Q0.0.2 (1)	%Q0.0.3 (1)
	Frequenzimetro	%I0.0.1	(2)	(2)	(2)	(2)	(2)

		Ingressi principali		Ingressi ausiliari		Uscite riflesse	
%VFC1	Uso selezionato	Ingresso IA	Ingresso IB)	IPres	Ica	Uscita 0	Uscita 1
	Contatore avanti/ indietro	%I0.0.7	%I0.0.6 (UP = 0/DO = 1)	%I0.0.5 (1)	%I0.0.4 (1)	%Q0.0.4 (1)	%Q0.0.5 (1)
	Contatore bifase avanti/indietro	%I0.0.7	%I0.0.6 (Impulso)	%I0.0.5 (1)	%I0.0.4 (1)	%Q0.0.4 (1)	%Q0.0.5 (1)
	Contatore avanti singolo	%I0.0.7	(2)	%I0.0.5 (1)	%I0.0.4 (1)	%Q0.0.4 (1)	%Q0.0.5 (1)
	Contatore indietro singolo	%I0.0.7	(2)	%I0.0.5 (1)	%I0.0.4 (1)	%Q0.0.4 (1)	%Q0.0.5 (1)
	Frequenzimetro	%I0.0.7	(2)	(2)	(2)	(2)	(2)

Commenti:

(1) = facoltativo

(2) = non utilizzato

Ipres = ingresso di preselezione

Ica= ingresso cattura

Ingresso IA = ingresso impulsi

Ingresso IB = impulsi o UP/DO

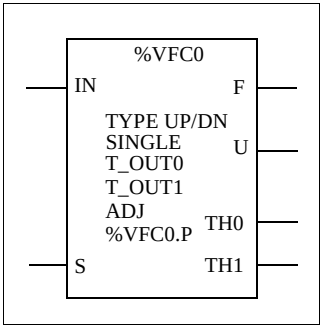
UP/DO = conteggio avanti/indietro

Se non utilizzato, l'ingresso o l'uscita rimane un I/O digitale normale disponibile per la gestione da parte dell'applicazione nel ciclo principale.

Se si utilizza %I0.0.2, %FC0 non è disponibile.
Se si utilizza %I0.0.3, %FC2 non è disponibile.
Se si utilizza %I0.0.4, %FC3 non è disponibile.

Illustrazione

Questa è una rappresentazione a blocchi del contatore molto veloce: (%VFC) in modalità a parola singola:



Specifiche tecniche La tabella seguente elenca le caratteristiche del blocco funzione Contatore molto veloce (%VFC).

Funzione	Descrizione	Valori	Uso del contatore molto veloce	Accesso in corso di esecuzione
Valore corrente (%VFCi.V) (%VFCi.VD*)	Valore corrente incrementato o diminuito in base agli ingressi fisici e alla funzione selezionata. Il valore può essere preimpostato o inizializzato tramite l'ingresso di preselezione. (%VFCi.S).	%VFCi.V: 0 -> 65535 %VFCi.VD: 0 -> 4294967295	CM	Lettura
Valore preimpostato (%VFCi.P) (%VFCi.PD*)	Utilizzato solo per la funzione di conteggio avanti/indietro e conteggio avanti/indietro singolo.	%VFCi.P: 0 -> 65535 %VFCi.PD: 0 -> 4294967295	CM o FM	Lettura e scrittura (1)
Valore di cattura (%VFCi.C) (%VFCi.CD*)	Utilizzato solo per la funzione di conteggio avanti/indietro e conteggio avanti/indietro singolo.	%VFCi.C: 0 -> 65535 %VFCi.CD: 0 -> 4294967295	CM	Lettura
Direzione conteggio (%VFCi.U)	Impostato dal sistema, questo bit viene utilizzato dalla funzione di conteggio avanti/indietro per indicare la direzione del conteggio: Come contatore avanti/indietro monofase, %I0.0.0 stabilisce la direzione di %VFC0 e %I0.0.6 per %VFC1. Per un contatore avanti/indietro bifase, è la differenza di fase tra i due segnali a determinare la direzione. Per %VFC0, %I0.0 è dedicato a IB e %I0.1 a IA. Per %VFC1, %I0.6 è dedicato a IB e %I0.7 a IA.	0 (Conteggio indietro) 1 (Conteggio avanti)	CM	Lettura
Attivazione uscita riflessa 0 (%VFCi.R)	Convalida uscita riflessa 0	0 (disabilita) 1 (abilita)	CM	Lettura e scrittura (2)
Attivazione uscita riflessa 1 (%VFCi.S)	Convalida uscita riflessa 1	0 (disabilita) 1 (abilita)	CM	Lettura e scrittura (2)
Valore soglia S0 (%VFCi.S0) (%VFCi.S0D*)	Questa parola contiene il valore soglia 0. Il significato viene definito durante la configurazione del blocco funzione. Nota: questo valore deve essere inferiore a %VFCi.S1.	%VFCi.S0: 0 -> 65535 %VFCi.S0D: 0 -> 4294967295	CM	Lettura e scrittura (1)
Valore soglia S1 (%VFCi.S1) (%VFCi.S1D*)	Questa parola contiene il valore soglia 0. Il significato viene definito durante la configurazione del blocco funzione. Nota: questo valore deve essere superiore a %VFCi.S0.	%VFCi.S1: 0 -> 65535 %VFCi.S1D: 0 -> 4294967295	CM	Lettura e scrittura (1)
Base tempo della misura di frequenza (%VFCi.T)	Voce configurabile della base tempo (100 o 1000 millisecondi).	1000 o 100	FM	Lettura e scrittura (1)

Funzione	Descrizione	Valori	Uso del contatore molto veloce	Accesso in corso di esecuzione
Regolabile (Y/N)	Voce configurabile che, quando selezionata, consente all'utente di modificare i valori di preimpostazione, soglia e base tempo della frequenza di misura durante l'esecuzione.	N (No) Y (Sì)	CM o FM	No
Ingresso di convalida (IN)	Utilizzato per convalidare o inibire la funzione corrente.	0 (No)	CM o FM	Lettura e scrittura (3)
Ingresso preset (S)	Dipende dalla configurazione allo stato 1: ● conteggio avanti/indietro o conteggio indietro: inizializza il valore corrente al valore preimpostato. ● conteggio avanti singolo: imposta a 0 il valore corrente: Inoltre, inizializza anche il funzionamento delle uscite di soglia e considera le modifiche dell'utente ai valori soglia impostati tramite il pannello visualizzatore o il programma utente.	0 o 1	CM o FM	Lettura e scrittura
Uscita overflow (F)	Da 0 a 65535 o da 65535 a 0 in modalità standard Da 0 a 4294967295 o da 4294967295 a 0 in modalità parola doppia	0 o 1	CM	Lettura
Soglia Bit 0 (%VFCi.TH0)	Impostato a 1 quando il valore corrente è maggiore o uguale al valore di soglia %VFCi.S0. È consigliabile provare questo bit solo una volta nel programma perché viene aggiornato in tempo reale. L'applicazione utente è responsabile della validità del valore al momento dell'uso.	0 o 1	CM	Lettura
Soglia Bit 1 (%VFCi.TH1)	Impostato a 1 quando il valore corrente è maggiore o uguale al valore di soglia %VFCi.S1. È consigliabile provare questo bit solo una volta nel programma perché viene aggiornato in tempo reale. L'applicazione utente è responsabile della validità del valore al momento dell'uso.	0 o 1	CM	Lettura

(*)Significa una variabile parola doppia a 32 bit. L'opzione parola doppia è disponibile su tutti i controller ad eccezione dei controller Twido TWDLC•A10DRF.
(1) Può essere scritto solo se Regolazione è impostata a 1.
(2) Accesso disponibile solo se configurato.
(3)Può essere scritto e letto solo dall'applicazione. Accesso impossibile dal pannello visualizzatore o dall'editor della tabella di animazione.
CM = Counting Mode (modalità di conteggio)
FM = Frequency Meter Mode (modalità frequenzimetro)

Descrizione della funzione di conteggio

La funzione di conteggio veloce (%VFC) funziona a una frequenza massima di 20 kHz, con un intervallo compreso tra 0 e 65535 in modalità standard e da 0 a 4294967295. Gli impulsi sono applicati come segue:
Tabella:

Funzione	Descrizione	%VFC0		%VFC1	
		IA	IB	IA	IB
Contatore avanti/ indietro	Gli impulsi sono applicati all'ingresso fisico, l'operazione corrente (conteggio avanti/indietro) viene definita dallo stato dell'ingresso fisico IB.	%I0.0.1	%I0.0.0	%I0.0.7	%I0.0.6
Contatore bifase avanti/indietro	Le due fasi dell'encoder vengono applicate agli ingressi fisici IA e IB.	%I0.0.1	%I0.0.0	%I0.0.7	%I0.0.6
Contatore avanti singolo	Gli impulsi vengono applicati all'ingresso fisico IA. IB non viene utilizzato.	%I0.0.1	NA	%I0.0.7	NA
Contatore indietro singolo	Gli impulsi vengono applicati all'ingresso fisico IA. IB non viene utilizzato.	%I0.0.1	NA	%I0.0.7	NA

Note sui blocchi funzione

Le operazioni di conteggio avanti o indietro vengono eseguite sul fronte di salita degli impulsi e solo se il blocco conteggio è abilitato.
Esistono due ingressi opzionali utilizzati nella modalità di conteggio: ICa e IPres. ICa viene utilizzato per catturare il valore corrente (%VFCi.V o %VFCi.VD) e memorizzarlo in %VFCi.C o %VFCi.CD. Gli ingressi Ica sono specificati come %I0.0.3 per %VFC0 e %I0.0.4 per %VFC1, se disponibile.
Quando l'ingresso IPres è attivo, il valore corrente viene modificato come segue:

- Per il conteggio in avanti, %VFCi.V o %VFCi.VD viene azzerato.
- Per il conteggio all'indietro, %VFCi.V o %VFCi.VD viene scritto con il contenuto rispettivamente di %VFCi.P o %VFCi.PD.
- Per il conteggio della frequenza, %VFCi.V o %VFCi.PD viene azzerato

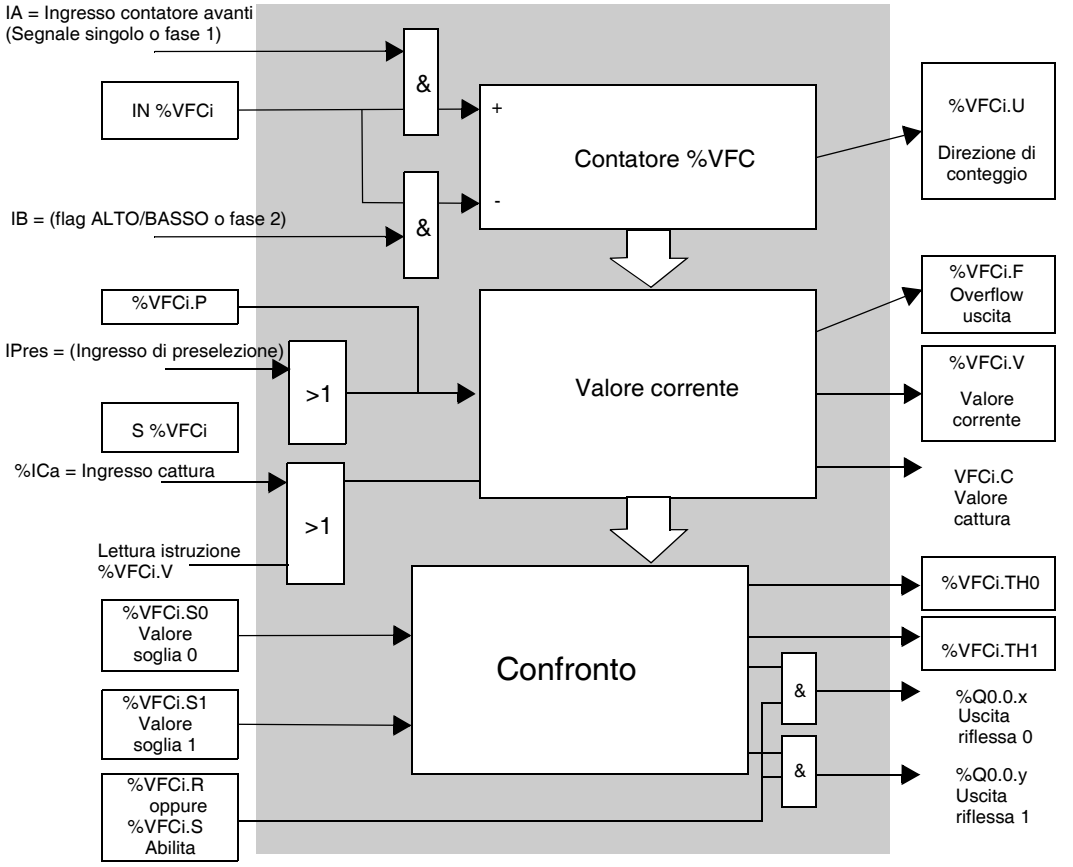
Attenzione: anche %VFCi.F è impostato a 0. Gli ingressi IPres sono specificati come %I0.0.2 per %VFC0 e %I0.0.5 per %VFC1 se disponibile.

Note sulle uscite dei blocchi funzione

Per tutte le funzioni, il valore corrente è confrontato con le due soglie (%VFCi.S0 o %VFCi.S0D e % VFCi.S1 o %VFCi.S1D). In base al risultato di questo confronto due oggetti bit (%VFCi.TH0 e %VFCi.TH1) vengono impostati a 1 se il valore corrente è maggiore o uguale alla soglia corrispondente o azzerato in caso contrario. Le uscite riflesse (se configurate) vengono impostate a 1 in base a questi confronti. Nota: è possibile configurare 1 o 2 uscite o nessuna uscita.
%VFC.U è un'uscita di FB, fornisce la direzione della variazione del contatore associato (1 per UP, 0 per DOWN).

Diagramma delle funzioni di conteggio

Di seguito è riportato un diagramma di funzione di conteggio in modalità standard (in modalità parola doppia si useranno le variabili funzione parola doppia):



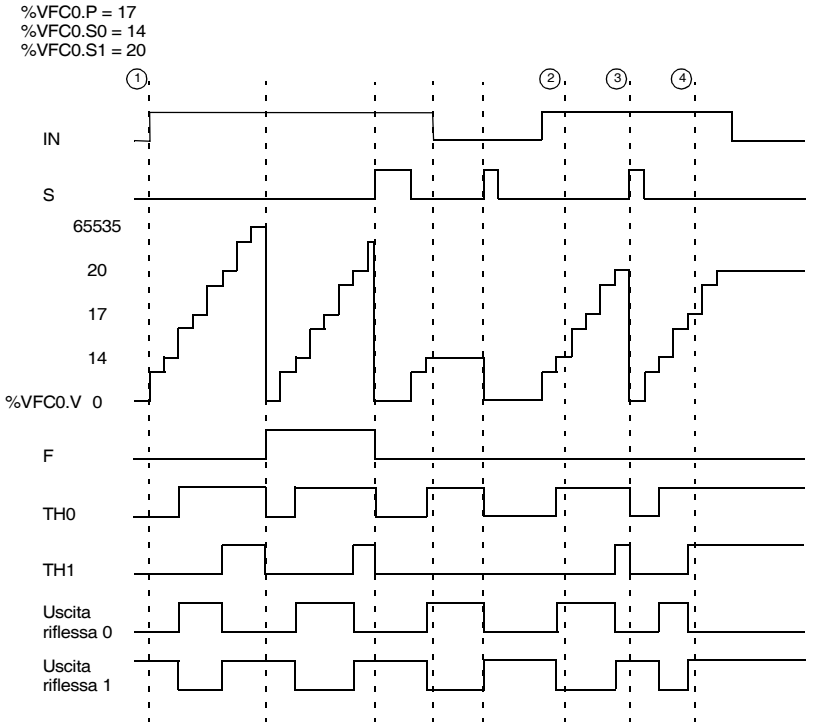
Nota: le uscite vengono gestite indipendentemente dal tempo di ciclo del controller. Il tempo di risposta è compreso tra 0 e 1 ms.

Operazione contatore avanti singolo

Di seguito viene riportato un esempio dell'uso di %VFC in modalità contatore avanti singolo. Per questo esempio sono stati impostati i seguenti elementi di configurazione:
Il valore preimpostato di %VFC0.P è 17%, mentre il valore soglia inferiore di %VFC0.S0 è 14 e il valore soglia superiore di %VFC0.S1 è 20.

Uscita riflessa	valore < %VFC.S0	%VFC0.S0 <= valore < %VFC0.S1	valore >= %VFC0.S1
%Q0.0.2		X	
%Q0.0.3	X		X

Di seguito è riportato un diagramma dei tempi:



(1) : %VFC0.U = 1 perché %VFC è un contatore in avanti

(2) : modifica di %VFC0.S1 a 17

(3) : l'attivazione dell'ingresso S conferma il nuovo valore della soglia S1 nel conteggio indietro successivo

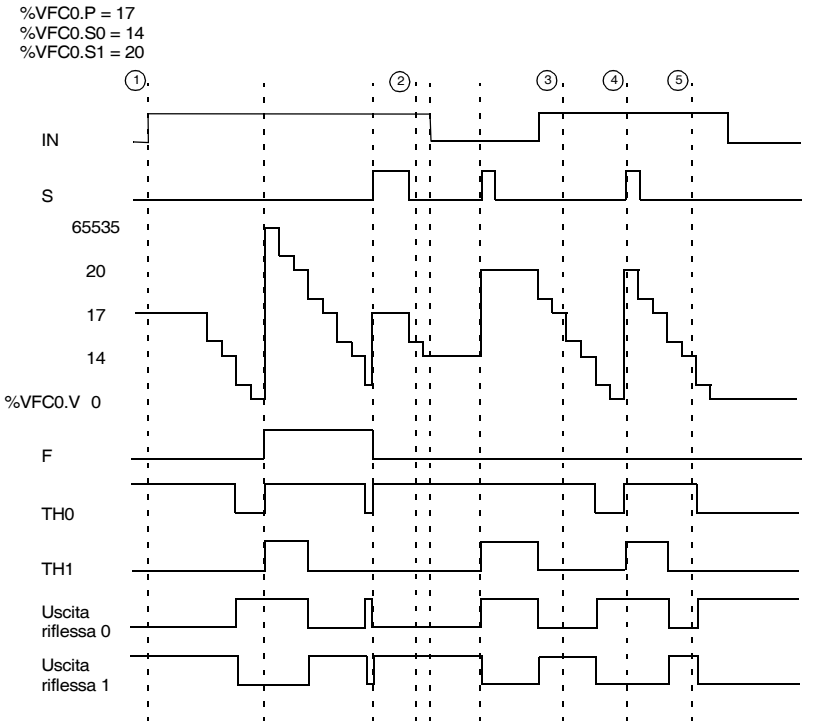
(4) : viene eseguito un catch del valore corrente, quindi %VFC0.C = 17

Operazione contatore indietro singolo

Di seguito viene riportato un esempio dell'uso di %VFC in modalità contatore indietro singolo. Per questo esempio sono stati impostati i seguenti elementi di configurazione:
Il valore preimpostato di %VFC0.P è 17%, mentre il valore soglia inferiore di %VFC0.S0 è 14 e il valore soglia superiore di %VFC0.S1 è 20.

Uscita riflessa	valore < %VFC.S0	%VFC0.S0 <= valore < %VFC0.S1	valore >= %VFC0.S1
%Q0.0.2	X		X
%Q0.0.3		X	

Esempio:



① : %VFC0.U = 0 perché %VFC è un contatore indietro

② : modifica di %VFC0.P a 20

③ : modifica di %VFC0.S1 a 17

④ : l'attivazione dell'ingresso S conferma il nuovo valore della soglia S1 nel conteggio indietro successivo

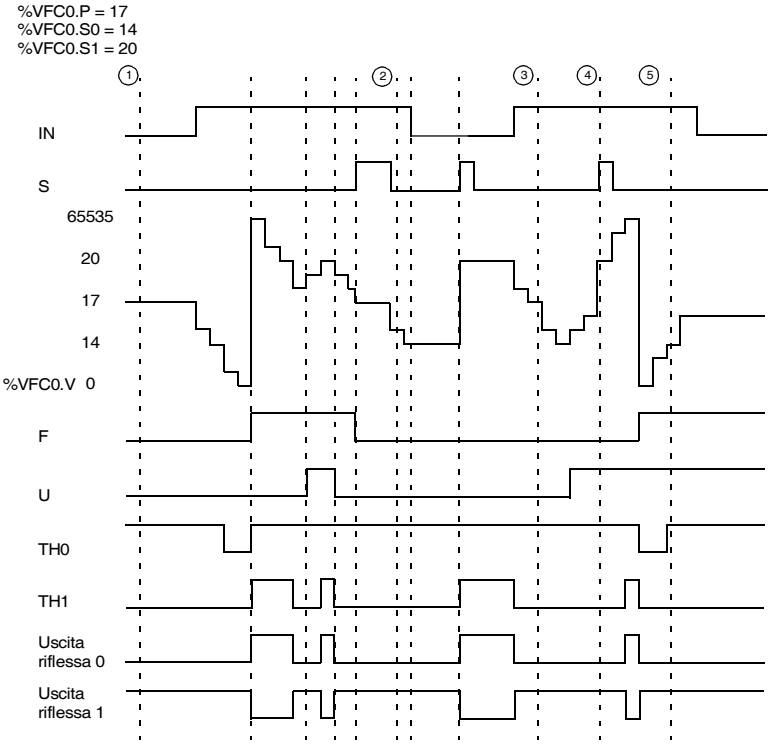
⑤ : viene eseguito un catch del valore corrente, quindi %VFC0.C = 17

Operazione contatore avanti/ indietro

Di seguito viene riportato un esempio dell'uso di %VFC in modalità contatore avanti/ indietro. Per questo esempio sono stati impostati i seguenti elementi di configurazione:
Il valore preimpostato di %VFC0.P è 17%, mentre il valore soglia inferiore di %VFC0.S0 è 14 e il valore soglia superiore di %VFC0.S1 è 20.

Uscita riflessa	valore < %VFC.S0	%VFC0.S0 <= valore < %VFC0.S1	valore >= %VFC0.S1
%Q0.0.2			X
%Q0.0.3	X	X	

Esempio:



① : ingresso IN impostato a 1 e ingresso S a 1

② : modifica di %VFC0.P a 20

③ : modifica di %VFC0.S1 a 17

④ : l'attivazione dell'ingresso S conferma il nuovo valore della soglia S1 nel conteggio indietro successivo

⑤ : viene eseguito un catch del valore corrente, quindi %VFC0.C = 17

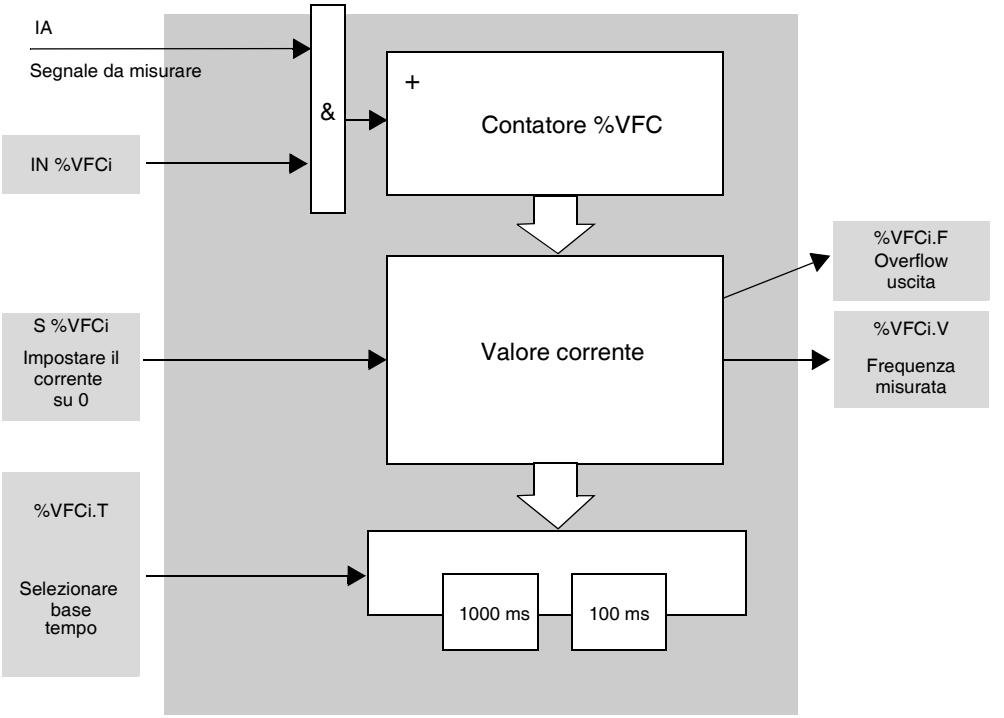
Descrizione della funzione frequenzimetro

La funzione frequenzimetro di un %VFC viene utilizzata per misurare la frequenza di un segnale periodico in Hz sull'uscita IA. L'intervallo di frequenza che può essere misurato va da 10 Hz a 20 kHz. L'utente può scegliere tra due basi tempo, e la scelta viene fatta tramite un nuovo oggetto %VFC.T (base tempo). Un valore di 100 corrisponde a una base tempo di 100 ms e un valore di 1000 a un base tempo di 1 secondo.

Base tempo	Intervallo di misurazione	Precisione	Aggiornamento
100 ms	da 100 Hz a 20 kHz	0,05 % per 20 kHz 10 % per 100 Hz	10 volte al secondo
1 s	da 10 Hz a 20 kHz	0,005 % per 20 kHz 10 % per 10 Hz	Una volta al secondo

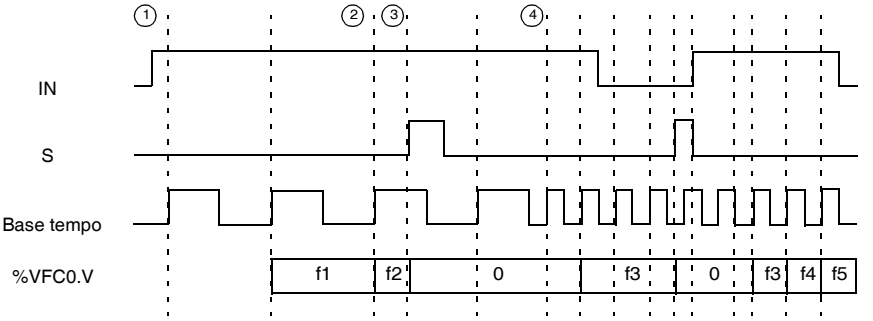
Diagramma della funzione frequenzimetro

Di seguito viene riportato un diagramma della funzione frequenzimetro:



Operazione di misurazione della frequenza

Di seguito viene fornito un esempio di diagramma dei tempi dell'utilizzo di %VFC in modalità frequenzimetro.



① : la misurazione della prima frequenza inizia qui.

② : il valore della frequenza corrente viene aggiornato.

③ : ingresso IN è impostato a 1 e ingresso S a 1

④ : modifica di %VFC0.T per 100 ms: questa modifica annulla la misurazione corrente e ne avvia un'altra.

Casi speciali

La tabella seguente contiene un elenco di casi speciali per il funzionamento del blocco funzione %VFC.

Caso speciale	Descrizione
Effetto di un avvio a freddo (%S0=1)	Reimposta tutti gli attributi %VFC con i valori configurati dall'utente o dall'applicazione utente.
Effetto di un riavvio a caldo (%S1=1)	Non ha alcun effetto.
Effetto di un arresto del controller	Il %VFC interrompe il funzionamento e le uscite rimangono nello stato corrente.

Trasmissione e ricezione di messaggi – Istruzione di scambio (EXCH)

Introduzione

Un controller Twido può essere configurato per comunicare con periferiche slave Modbus o inviare e ricevere messaggi in modalità carattere (ASCII).
Per queste comunicazioni TwidoSoft fornisce le seguenti funzioni:

- Istruzione EXCH per trasmettere/ricevere i messaggi
- Blocco funzione di controllo di scambio %MSG per controllare gli scambi di dati

Il controller Twido utilizza il protocollo configurato per la porta specifica durante l'elaborazione di un'istruzione EXCH. A ogni porta di comunicazione è possibile assegnare un protocollo diverso. L'accesso alle porte di comunicazione avviene aggiungendo il numero della porta alla funzione EXCH o %MSG (EXCH1, EXCH2, %MSG1, %MSG2).
Inoltre, i controller di serie TWDLCAE40DRF implementano la messaggeria Modbus TCP tramite la rete Ethernet usando l'istruzione EXCH3 e la funzione %MSG3.

EXCH, istruzione

L'istruzione EXCH consente a un controller Twido di inviare o ricevere informazioni da/a periferiche ASCII. L'utente definisce una tabella di parole (%MWi:L) contenente i dati da inviare o ricevere (fino a 250 byte di dati in trasmissione e/o ricezione). Il formato delle tabelle di parole è descritto nei paragrafi relativi a ciascun protocollo. Uno scambio di messaggi viene eseguito utilizzando l'istruzione EXCH.

Sintassi

La sintassi da utilizzare per l'istruzione EXCH è indicato di seguito:
[EXCHx %MWi:L]
dove: x = numero di porta seriale (1 o 2); x = porta Ethernet (3); L = numero totale di parole della tabella di parole (massimo 121). Valori della tabella di parole interne %MWi:L sono ad esempio i+L <= 255.
Il controller Twido deve concludere lo scambio relativo alla prima istruzione EXCHx prima di poter avviare una seconda istruzione di scambio. Nel caso di invio di più messaggi, utilizzare il blocco funzione %MSG.

Nota: per maggiori informazioni sull'istruzione EXCH3 della messaggeria Modbus TCP, vedere *Messaggeria TCP Modbus, p. 185*.

Blocco funzione controllo di scambio (%MSGx)

Introduzione

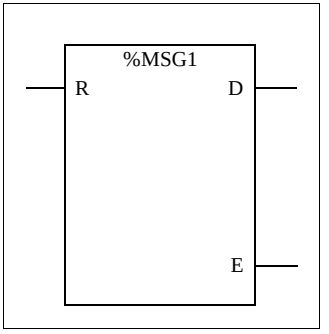
Nota: la "x" di %MSGx indica la porta del controller: "x = 1 or 2"
- x = 1 o 2, significa rispettivamente la porta seriale 1 o 2 del controller;
- x = 3, significa la porta di rete Ethernet del controller (solo sui controller TWDLCAE40DRF). Per maggiori informazioni sulla funzione %MSG3, vedere *Messaggeria TCP Modbus, p. 185*.

Il blocco funzione %MSGx gestisce gli scambi di dati ed ha tre funzioni:
- Controllo degli errori di comunicazione:
  Il controllo degli errori verifica che la lunghezza del blocco (tabella di parole) programmata con l'istruzione EXCH sia sufficientemente estesa per contenere il messaggio da inviare (confronto con la lunghezza programmata nel byte meno significativo della prima parola nella tabella di parole).
- Coordinamento di più messaggi:
  Per garantire il coordinamento quando si inviano più messaggi, il blocco funzione %MSGx contiene le informazioni che consentono di assicurarsi del corretto esito della trasmissione di ogni messaggio.
- Trasmissione di messaggi di priorità:
  Il blocco funzione %MSGx consente di interrompere la trasmissione del messaggio corrente per consentire l'invio immediato di un messaggio urgente.

La programmazione del blocco funzione %MSGx è opzionale.

Illustrazione

Di seguito viene illustrato il blocco funzione %MSGx.



Parametri

La tabella seguente elenca i parametri del blocco funzione %MSGx.

Parametro	Etichetta	Valore
Azzera ingresso (o istruzione)	R	Allo stato 1, viene reinizializzata la comunicazione: %MSGx.E = 0 e %MSGx.D = 1.
Uscita comunicazione terminata	%MSGx.D	Stato 1, comunicazione terminata se: ● fine della trasmissione (se esiste la trasmissione) ● fine della ricezione (ricevuto carattere di fine) ● Errore ● azzera il blocco Stato 0, richiesta in corso.
Uscita difettosa (errore)	%MSGx.E	Stato 1, comunicazione terminata se: ● comando errato ● configurazione tabella errata ● ricezione carattere errato (velocità, parità, ecc.) ● tabella di ricezione piena (non aggiornata) Stato 0, lunghezza messaggio corretta, collegamento corretto.

Se si verifica un errore durante l'esecuzione di un'istruzione EXCH, i bit %MSGx.D e %MSGx.E vengono impostati a 1, la parola di sistema %SW63 contiene il codice di errore della porta 1 e la parola di sistema %SW64 quello della porta 2. Vedere *Parole di sistema (%SW), p. 614*.

Azzera ingresso (R)

Quando Azzera ingresso è impostato a 1:
- La trasmissione di tutti i messaggi viene interrotta.
- L'uscita guasta (errore) viene azzerata.
- Il bit di conferma è impostato a 1.

Ora è possibile inviare un nuovo messaggio.

Uscita guasta (errore) (%MSGx.E)

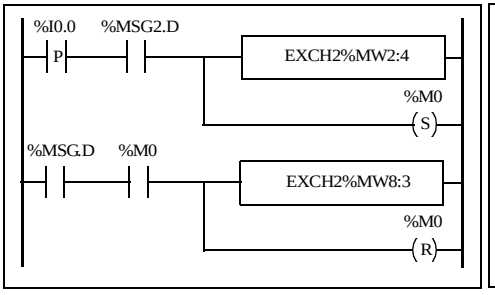
L'uscita guasta è impostata a 1 in caso di errore di programmazione delle comunicazioni o di errore di trasmissione del messaggio. L'uscita guasta è impostata a 1 se il numero di byte definiti nel blocco dati associato all'istruzione EXCH (parola 1, byte meno significativo) è maggiore di 128 (+80 in esadecimale tramite FA).
Anche l'uscita guasta viene impostata a 1 in caso di problema di invio di un messaggio Modbus a una periferica Modbus. In questo caso, l'utente deve controllare il cablaggio e che la periferica di destinazione supporti la comunicazione Modbus.

Uscita Comunicazione terminata (%MSGx.D)

Se l'uscita Comunicazione terminata è impostata a 1, il controller Twido è pronto per l'invio di un altro messaggio. L'utilizzo dell'uscita %MSGx.D è consigliato nel caso di invio di più messaggi. Se non viene utilizzata, i messaggi possono andare perduti.

Trasmissione di più messaggi successivi

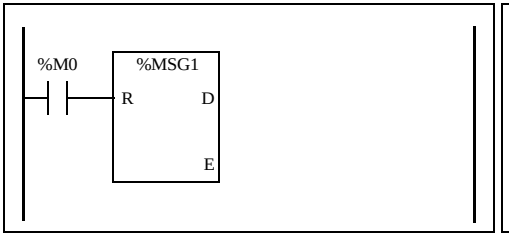
L'esecuzione dell'istruzione EXCH attiva un blocco messaggio nel programma applicativo. Il messaggio viene trasmesso se il blocco messaggio non è già attivo (%MSGx.D = 1). Se nello stesso ciclo vengono inviati più messaggi, viene trasmesso solo il primo. L'utente è responsabile della gestione della trasmissione di più messaggi utilizzando il programma.
Esempio di trasmissione di due messaggi in successione sulla porta 2:




```
LDR     %I0.0
AND
%MSG2.D
[EXCH2
%MW2:4]
S       %M0
LD
```

Reinizializ-zazione degli scambi

Uno scambio viene annullato attivando l'ingresso (o l'istruzione) R. Questo ingresso inizializza la comunicazione e imposta l'uscita MSGx.E e l'uscita %MSGx.D a 1. È possibile reinizializzare uno scambio se viene rilevato un errore.
Esempio di reinizializzazione di uno scambio:




```
BLK     %MSG1
LD      %M0
R
END_BLK
```

Casi speciali

La tabella seguente contiene i casi speciali di funzionamento del blocco funzione %MSGx.

Caso speciale	Descrizione
Effetto di un avvio a freddo (%S0=1)	Forza la reinizializzazione della comunicazione.
Effetto di un riavvio a caldo (%S1=1)	Non ha alcun effetto.
Effetto di un arresto del controller	Se è in corso la trasmissione di un messaggio, il controller blocca il trasferimento e reinizializza le uscite %MSGx.D e %MSGx.E.

17.2 Funzioni dell'orologio

In breve

Oggetto di questa sezione

Questa sezione descrive le funzioni di gestione dell'ora per i controller Twido.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Funzioni dell'orologio	484
Blocchi orologio-datario	485
Timbro data/ora	488
Impostazione della data e dell'ora	490

Funzioni dell'orologio

Introduzione

I controller Twido dispongono di una funzione di orologio che richiede l'opzione RTC (orologio in tempo reale) e mette a disposizione:

- **Blocchi di programmazione** utilizzati per controllare le azioni a ore predefinite o calcolate.
- **Timbro data/ora** utilizzato per assegnare ora e data a eventi e misurarne la durata.

È possibile accedere all'orologio in tempo reale Twido selezionando **Blocchi di programmazione** dal menu **Software** di TwidoSoft. Inoltre, l'orologio in tempo reale può essere impostato da programma. Le impostazioni dell'orologio continuano a funzionare per un massimo di 30 giorni dopo lo spegnimento del controller, se la batteria è stata caricata per almeno sei ore consecutive prima che il controller venga spento.

L'orologio in tempo reale ha il formato 24 ore e considera gli anni bisestili.

Valore di correzione RTC

Il valore di correzione RTC è necessario per il corretto funzionamento dell'orologio in tempo reale. Ciascuna unità RTC ha un valore di correzione proprio scritto nell'unità stessa. Questo valore è configurabile in TwidoSoft utilizzando l'opzione **Configura RTC** della finestra di dialogo **Operazioni del controller**.

Blocchi orologio-datario

Introduzione

I blocchi orologio-datario vengono utilizzati per controllare le azioni in un mese, giorno e ora predefiniti. È possibile programmare fino a 16 blocchi orologio-datario. Tali blocchi non richiedono alcuna immissione da programma.

Nota: verificare il bit di sistema %S51 a la parola di sistema %SW118 per assicurarsi che sia installata l'opzione orologio in tempo reale (RTC). Vedere *Bit di sistema (%S), p. 604*. L'opzione RTC è necessaria per l'uso dei blocchi orologio-datario.

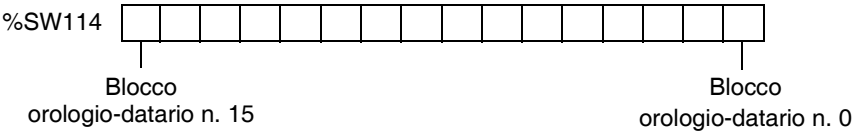
Impostazioni

La tabella seguente elenca i parametri di un blocco orologio-datario:

Parametro	Formato	Funzione/Intervallo
Numero del blocco orologio-datario	n	n = da 0 a 15
Configurato	Configurato	Selezionare questa casella per configurare il numero di blocco orologio-datario selezionato.
Bit d'uscita	%Qx.y.z	L'assegnazione dell'uscita è attivata dal blocco orologio-datario: %Mi o %Qj.k. Questa uscita è impostata a 1 quando la data e l'ora correnti sono compresi tra l'impostazione dell'inizio e della fine del periodo attivo.
Mese d'inizio	da gennaio a dicembre	Il mese di inizio del blocco orologio-datario.
Mese di fine	da gennaio a dicembre	Il mese di fine del blocco orologio-datario.
Data d'inizio	1 - 31	Il giorno del mese di inizio del blocco orologio-datario.
Data di fine	1 - 31	Il giorno del mese di fine del blocco orologio-datario.
Ora d'inizio	hh:mm	L'ora di inizio del blocco orologio-datario. Definita dall'ora (da 0 a 23), seguita dai minuti (da 0 a 59).
Ora di fine	hh:mm	L'ora di fine del blocco orologio-datario. Definita dall'ora (da 0 a 23), seguita dai minuti (da 0 a 59).
Giorno della settimana	Lunedì - Domenica	Caselle di controllo che identificano il giorno della settimana per l'attivazione del blocco orologio-datario.

Abilitazione dei blocchi orologio-datario

I bit della parola di sistema %SW114 abilitano (se impostati a 1) o disabilitano (se impostati a 0) il funzionamento di ciascuno dei 16 blocchi orologio-datario. Assegnazione dei blocchi orologio-datario in %SW114:



Per default (o dopo un avvio a freddo) tutti i bit di questa parola di sistema sono impostati a 1. L'utilizzo di questi bit da parte del programma è opzionale.

Uscita di blocchi orologio-datario

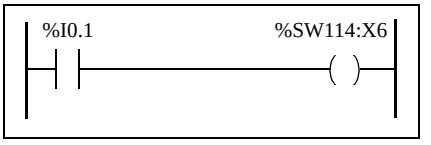
Se la stessa uscita (%Mi o %Qj.k) è assegnata da vari blocchi, l'istruzione OR dei risultati di ciascun blocco viene assegnata alla fine a questo oggetto (è possibile avere vari "intervalli operativi" per la stessa uscita).

Esempio

La tabella seguente mostra i parametri di un programma di irrigazione per un mese estivo:

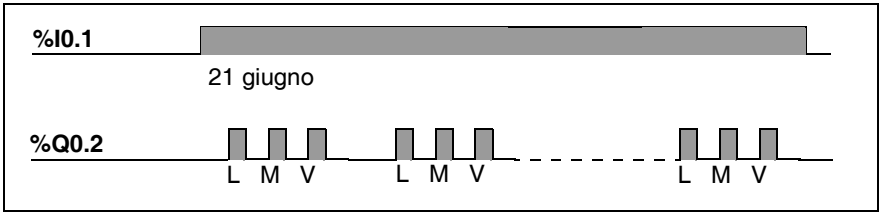
Parametro	Valore	Descrizione
Blocco orologio-datario	6	Blocco orologio-datario numero 6
Bit d'uscita	%Q0.2	Attivare l'uscita %Q0.2
Mese d'inizio	Giugno	Avviare l'attività in giugno
Mese di fine	Settembre	Interrompere l'attività in settembre
Data d'inizio	21	Avviare l'attività il 21 giugno
Data di fine	21	Interrompere l'attività il 21 settembre
Giorno della settimana	Lunedì, Mercoledì, Venerdì	Eseguire l'attività il lunedì, il mercoledì e il venerdì
Ora d'inizio	21:00	Avviare l'attività alle 21:00
Ora di fine	22:00	Interrompere l'attività alle 22:00

Utilizzando il seguente programma, il blocco orologio-datario può essere disabilitato mediante un interruttore o un rilevatore di umidità collegato all'ingresso %I0.1.




```
LD    %I0.1
ST    %SW114:X6
```

Il seguente diagramma dei tempi mostra l'attivazione dell'uscita %Q0.2.



Indicazione di data e ora tramite il programma

Data e ora sono disponibili nelle parole di sistema da %SW50 a %SW53 (consultare *Parole di sistema (%SW), p. 614*). Pertanto è possibile eseguire il timbro di data e ora nel programma del controller mediante confronti aritmetici tra la data e l'ora correnti e i valori immediati o le parole %MWi (o %KWi), che contengono i setpoint.

Timbro data/ora

Introduzione

Le parole di sistema da %SW49 a %SW53 contengono i parametri di data e di ora nel formato BCD (vedere la sezione *Analisi del codice BCD, p. 431*), utile per la visualizzazione o la trasmissione a una periferica. Queste parole di sistema possono essere utilizzate per memorizzare la data e l'ora di un evento (vedere *Parole di sistema (%SW), p. 614*.

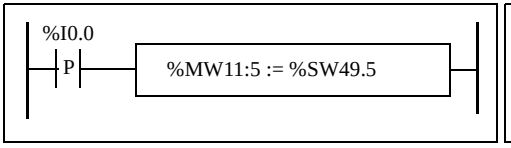
Nota: per impostare i parametri di data e di ora, inoltre, è possibile utilizzare il visualizzatore opzionale (vedere la sezione *Orologio in tempo reale, p. 323*).

Datazione di un evento

Per datare un evento è sufficiente utilizzare le operazioni di assegnazione per trasferire il contenuto delle parole di sistema alle parole interne, quindi elaborare queste ultime (ad esempio, la trasmissione all'unità visualizzatore tramite l'istruzione EXCH).

Esempio di programmazione

L'esempio che segue mostra come datare un fronte di salita sull'ingresso %I0.1.




```
LDR      %I0.0
[%MW11:5 := %SW49:5]
```

Dopo aver rilevato l'evento, la tabella di parole contiene:

Codifica	Byte più significativo	Byte meno significativo
%MW11		Giorno della settimana[1]
%MW12	00	Secondo
%MW13	Ora	Minuto
%MW14	Mese	Giorno
%MW15	Secolo	Anno

Nota: (1) 1 = Lunedì, 2 = Martedì, 3 = Mercoledì, 4 = Giovedì, 5 = Venerdì, 6 = Sabato, 7 = Domenica.

Esempio di tabella di parole

Esempio di data per lunedì, 19 Aprile, 2002, alle ore 13:40:30:

Parola	Valore (hex)	Significato
%MW11	0001	Lunedì
%MW12	0030	30 secondi
%MW13	1340	ore 13, 40 minuti
%MW14	0419	04 = 19 Aprile
%MW15	2002	2002

Data e ora dell'ultimo arresto

Le parole di sistema da %SW54 a %SW57 contengono i parametri di data e di ora dell'ultimo arresto e la parola %SW58 contiene il codice che ne visualizza la causa, nel formato BCD (vedere la sezione *Parole di sistema (%SW), p. 614*).

Impostazione della data e dell'ora

Introduzione

Le impostazioni di data e ora possono essere aggiornate tramite uno dei seguenti metodi:

- TwidoSoft
  Finestra di dialogo **Imposta l'ora**. È possibile accedervi dalla finestra di dialogo **Operazioni del Controller**. Per visualizzarla, selezionare **Verifica controller** nel menu **Controller**.
- Parole di sistema
  Utilizzare le parole di sistema da %SW49 a %SW53 o la parola di sistema %SW59.

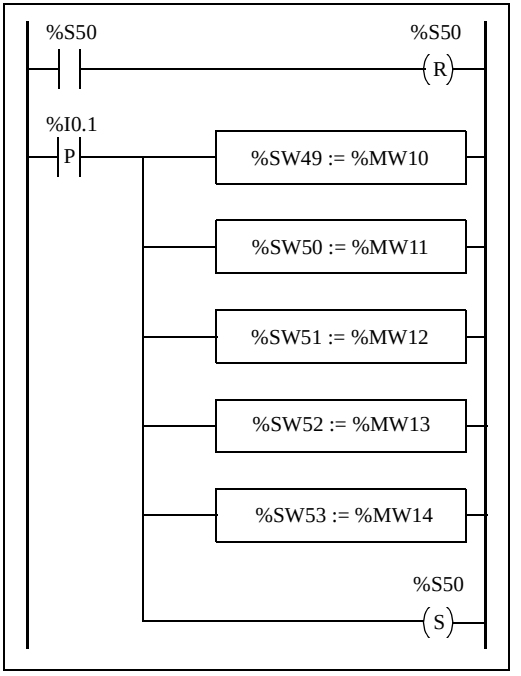
Le impostazioni di data e ora possono essere aggiornate solo quando la cartuccia RTC (TWDXCPRTC) opzionale è installata sul controller. Si osservi che nella serie di controller compatti TWDLCA•40DRF la cartuccia RTC è preinstallata.

Uso delle parole di sistema da %SW49 a %SW53

Per utilizzare le parole di sistema da %SW49 a %SW53 per impostare data e ora, il bit %S50 deve essere impostato a 1, cosa che determina quanto segue.

- Annullamento dell'aggiornamento delle parole da %SW49 a %SW53 tramite l'orologio interno.
- Trasmissione dei valori scritti nelle parole da %SW49 a %SW53 all'orologio interno.

Esempio di programmazione:




```
LD      %S50
R       %S50

LDR     %I0.1
[%SW49 := %MW10]
[%SW50 := %MW11]
[%SW51 := %MW12]
[%SW52 := %MW13]
[%SW53 := %MW14]
S       %S50
```

Le parole da %MW10 a %MW14 conterranno nuovi valori di data e ora in formato BCD (vedere *Analisi del codice BCD, p. 431*) e corrisponderanno alla codifica delle parole da %SW49 a %SW53.
La tabella di parole deve contenere i nuovi valori di data e ora.

Codifica	Byte più significativo	Byte meno significativo
%MW10		Giorno della settimana[1]
%MW11		Secondo
%MW12	Ora	Minuto
%MW13	Mese	Giorno
%MW14	Secolo	Anno

Nota: (1) 1 = Lunedì, 2 = Martedì, 3 = Mercoledì, 4 = Giovedì, 5 = Venerdì, 6 = Sabato, 7 = Domenica.

Esempio di data per Lunedì, 19 Aprile, 2002:

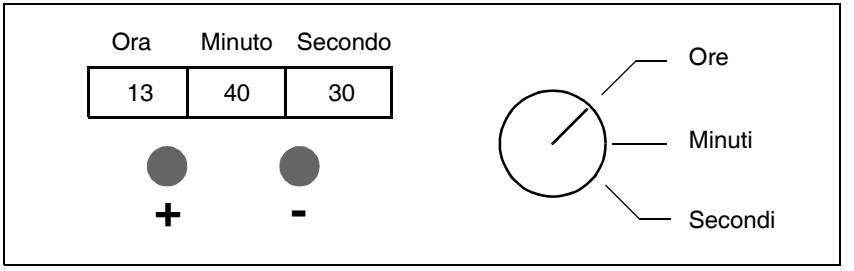
Parola	Valore (hex)	Significato
%MW10	0001	Lunedì
%MW11	0030	30 secondi
%MW12	1340	ore 13, 40 minuti
%MW13	0419	04 = 19 Aprile
%MW14	2002	2002

Utilizzo della parola %SW59

Un altro metodo per aggiornare la data e l'ora consiste nell'uso del bit di sistema %S59 e della parola di sistema %SW59.
L'impostazione del bit %S59 a 1 consente la regolazione della data e dell'ora corrente tramite la parola %SW59 (vedere *Parole di sistema (%SW), p. 614*). La parola %SW59 incrementa o decrementa ciascun componente di data e ora su un fronte di salita.

Esempio di applicazione

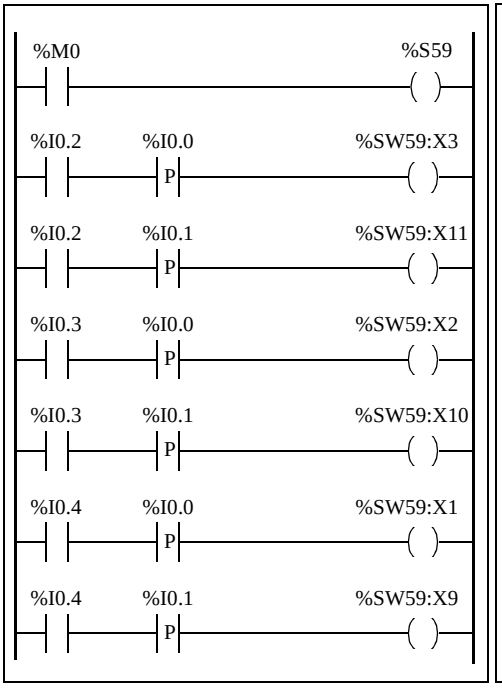
Il seguente pannello frontale viene creato per modificare ora, minuti e secondi di un orologio interno.



Descrizione dei comandi:

- L'interruttore Ore/Minuti/Secondi seleziona la visualizzazione dell'ora da cambiare utilizzando gli ingressi %I0.2, %I0.3 e %I0.4 rispettivamente.
- Il pulsante "+" incrementa la visualizzazione dell'ora selezionata utilizzando l'ingresso %I0.0.
- Il pulsante "-" decrementa la visualizzazione dell'ora selezionata utilizzando l'ingresso %I0.1.

Il programma seguente legge gli ingressi dal pannello e imposta l'orologio interno.




```
LD    %M0
ST    %S59
LD    %I0.2          (Ora)
ANDR  %I0.0
ST    %SW59:X3
LD    %I0.2
ANDR  %I0.1
ST    %SW59:X11
LD    %I0.3          (Minuto)
ANDR  %I0.0
ST    %SW59:X2
LD    %I0.3
ANDR  %I0.1
ST    %SW59:X10
LD    %I0.4          (Secondo)
ANDR  %I0.0
ST    %SW59:X1
LD    %I0.4
ANDR  %I0.1
ST    %SW59:X9
```

17.3 Guida all'avvio rapido della funzione PID Twido

In breve

Panoramica

Questa sezione contiene informazioni introduttive sulle funzioni di controllo PID e di sintonizzazione automatica disponibili sui controller Twido.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Scopo del documento	494
Fase 1 - Configurazione dei canali analogici utilizzati per la regolazione	496
Fase 2 - Prerequisiti della configurazione del PID	498
Fase 3 - Configurazione del PID	500
Fase 4 - Inizializzazione dell'implementazione della regolazione	507
Fase 5 - Implementazione della regolazione AT + PID	511
Fase 6 - Debug delle regolazioni	515

Scopo del documento

Introduzione

Questa guida di avvio rapido ha per scopo di guidare l'utente, con un esempio di utilizzo, attraverso tutte le fasi di configurazione e d'implementazione delle funzioni di regolazione PID del controller Twido.

Nota: l'implementazione della funzione PID di Twido non richiede delle conoscenze specifiche, ma richiede molto rigore nell'esecuzione delle operazioni per ottenere il migliore risultato nel minor tempo possibile.

Questa guida contiene

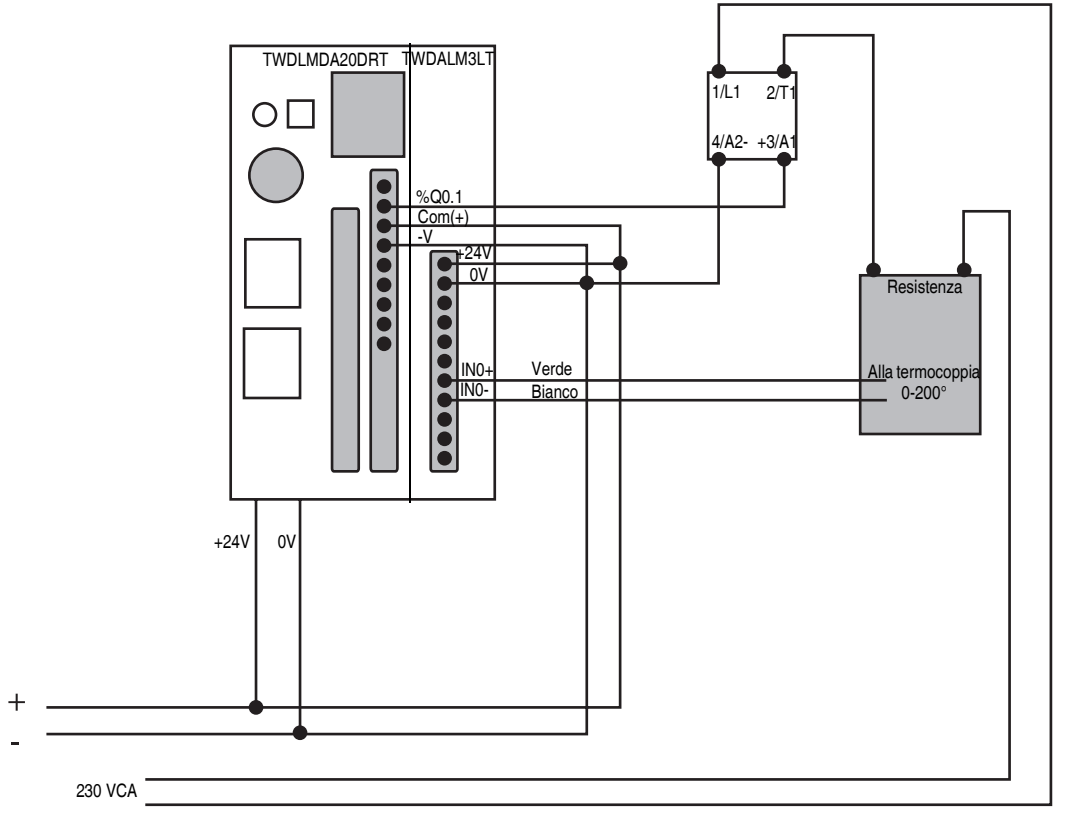
Il documento sviluppa le seguenti fasi:

Fase	Descrizione
1	Configurazione dei canali analogici utilizzati per la regolazione
2	Prerequisiti di configurazione del PID
3	Configurazione del PID
4	Inizializzazione della messa opera della regolazione
5	Implementazione della regolazione AT + PID
6	Debugging e regolazioni

Informazioni sull'esempio utilizzato in questa guida

In questo esempio, si è scelto di utilizzare una termocoppia Type K, 0-200°. Verrà utilizzato un sistema di governo con relè statico; l'uscita è quella del controller di base comandata in PWM direttamente dal regolatore PID (vedere *Fase 3 - Configurazione del PID, p. 500*

Lo schema qui di seguito rappresenta la costruzione sperimentale utilizzata in questo esempio di applicazione:



Fase 1 - Configurazione dei canali analogici utilizzati per la regolazione

Introduzione

In generale, un regolatore PID si serve di un segnale analogico di ritorno (chiamato "valore di misura") per misurare il valore da regolare.
Questo valore, può rappresentare un livello, una temperatura, una distanza o altro, a seconda delle applicazioni.

Esempio di segnale analogico di misura

L'esempio utilizza una misura della temperatura.
Il sensore utilizzato invia al controller una misura analogica che in funzione del valore misurato. Per la temperatura, e con i sensori i PT100 o le termocoppie, il segnale misurato aumenta con l'aumentare della temperatura.

Aggiunta di una scheda analogica (modulo d'estensione)

In modalità locale, una volta che è stata scelta la base del controller, aggiungere una scheda analogica come estensione della base. Il rilevamento dei canali della scheda dipenderà dal suo posizionamento nella configurazione.

Configurazione dei canali analogici

La seguente tabella indica la procedura da seguire per configurare i canali analogici del modulo d'estensione:

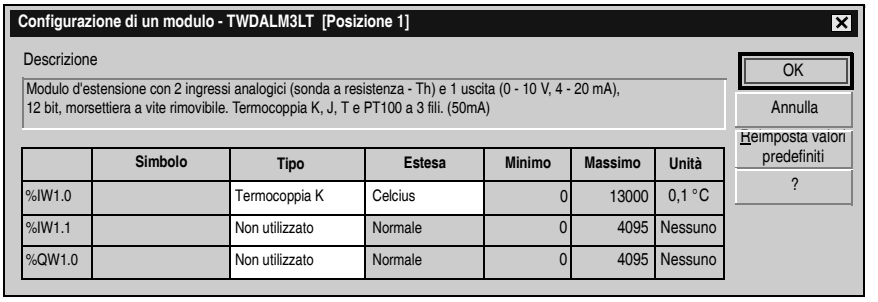
Fase	Azione
1	Fare clic con il pulsante destro sull'elemento **Bus di espansione** → **Aggiungi un modulo**.
2	Scegliere dall'elenco la scheda desiderata. Ad esempio, TWDALM3LT per una misura della temperatura con PT100 o una termocoppia.
3	Fare clic su **Aggiungi** poi su **OK** se la configurazione si limita a questo solo modulo di espansione.
4	Fare clic sulla scheda aggiunta e poi sul menu **Configura**.
5	Nella colonna **Tipo**, scegliere il tipo d'ingresso in funzione del sensore utilizzato (ThermoCouple K, se il sensore è di questo tipo).
6	Nella colonna **Estesa**, scegliere l'unità di misura del sensore. Per i sensori di temperatura è più facile scegliere **Celsius**, che permette di avere un rapporto diretto tra il numero di punti inviati dalla scheda analogica e la misura reale.
7	Rilevare il simbolo dell'ingresso della scheda analogica così configurata. Serve a completare i campi del PID (%IW1.0, per questo esempio).
8	Procedere allo stesso modo per un'uscita analogica se quest'ultima deve essere utilizzata per governare il sistema di comando.

Esempio di configurazione del canale analogico.

Sono possibili più tipi di configurazione secondo il tipo di misura attuata, come indicato qui di seguito:

- Nel caso dell'esempio dell'applicazione trattata in questo documento, si è scelta una **ThermoCouple Type K**, 0-200°. La lettura del valore della misura sarà direttamente comprensibile (2000 pts = 200° in quanto il rapporto dell'unità è di 0,1°).
- Per altri tipi di misura, si può scegliere **0-10V** o **4-20 mA** nella colonna **Tipo**, nonché **Personalizzato** nella colonna **Esteso**. Regolare la scala dei valori (immettere **0** nella colonna **Min.**, e **10000** nella colonna **Max.**) per avere una lettura diretta del valore di misura (10 V = 10000 punti).

L'esempio qui di seguito illustra la configurazione di un canale analogico di tipo ThermoCouple K:



	Simbolo	Tipo	Estesa	Minimo	Massimo	Unità
%IW1.0		Termocoppia K	Celcius	0	13000	0,1 °C
%IW1.1		Non utilizzato	Normale	0	4095	Nessuno
%QW1.0		Non utilizzato	Normale	0	4095	Nessuno

Fase 2 - Prerequisiti della configurazione del PID

Introduzione

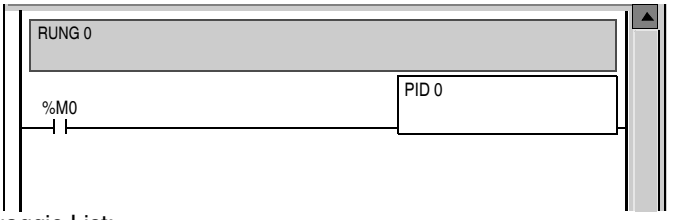
Prima di configurare il PID, assicurarsi che le seguenti fasi siano state realizzate:

Fase	Descrizione
1	Convalida del PID nel programma
2	Configurazione del periodo di scansione

Convalida del PID nel programma

Il regolatore PID deve essere convalidato nel programma con un'istruzione. Questa istruzione può essere permanente o condizionata con un ingresso o con un bit interno. Nell'esempio che segue, il PID è convalidato con l'istruzione %M0:

- In linguaggio Ladder:



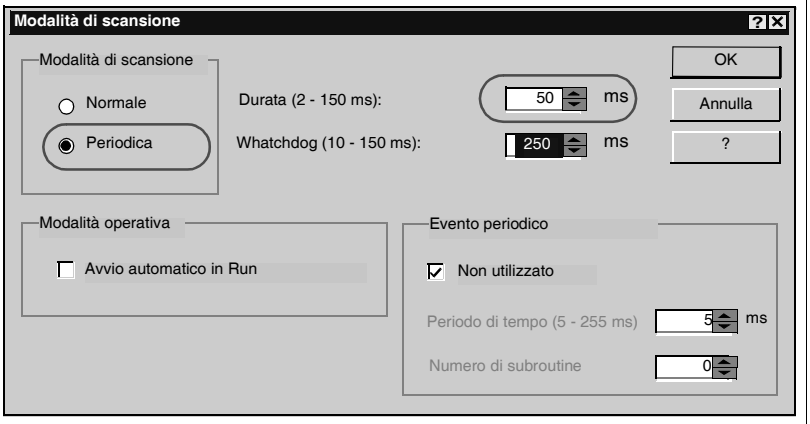
- In linguaggio List:

```
----
   0 LD     %M0
   1 [ PID 0 ]
```

Nota: Attenzione alla sintassi.
Verificare la presenza dello spazio tra i caratteri "PID" e il numero di PID (come `PID<espace>0`).

Configurazione del periodo di scansione

Durante l'utilizzo dei regolatori PID, si consiglia di configurare la modalità di scansione del ciclo controller in "periodica". La tabella che segue espone la procedura da seguire per configurare la modalità di scansione.

Fase	Azione
1	Nella barra dei menu di TwidoSoft, scegliere **Programma** → **Modifica modalità di scansione**.
2	Spuntare la casella **Periodica**.
3	Regolare il tempo di ciclo come indicato nella schermata di seguito:  **Nota:** Il tempo di ciclo deve essere adattato in funzione della dimensione del programma e delle prestazioni ricercate. (Il tempo di 50 ms è in molti casi un buon compromesso).

Fase 3 - Configurazione del PID

Introduzione

Per questo esempio è stato scelto di implementare la maggior parte delle funzioni del regolatore PID per Twido. Alcune scelte non sono indispensabili e possono essere semplificate.

Funzione Auto-Tuning (AT)

Il regolatore PID dispone di una funzione Auto-Tuning che permette di semplificare la regolazione del loop di regolazione (funzione definita AT nel presente documento).

Modalità operative

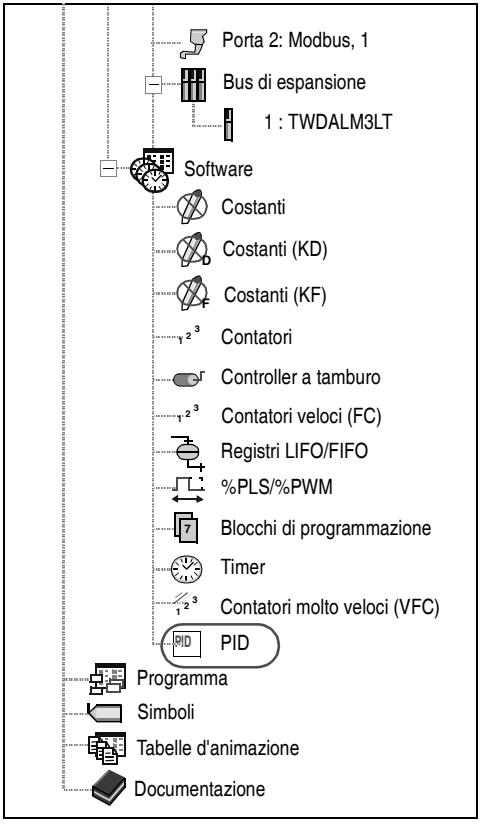
Il regolatore PID del controller Twido dispone di quattro modalità di funzionamento distinte, configurabili nella scheda **Generale** della finestra di dialogo **PID**:

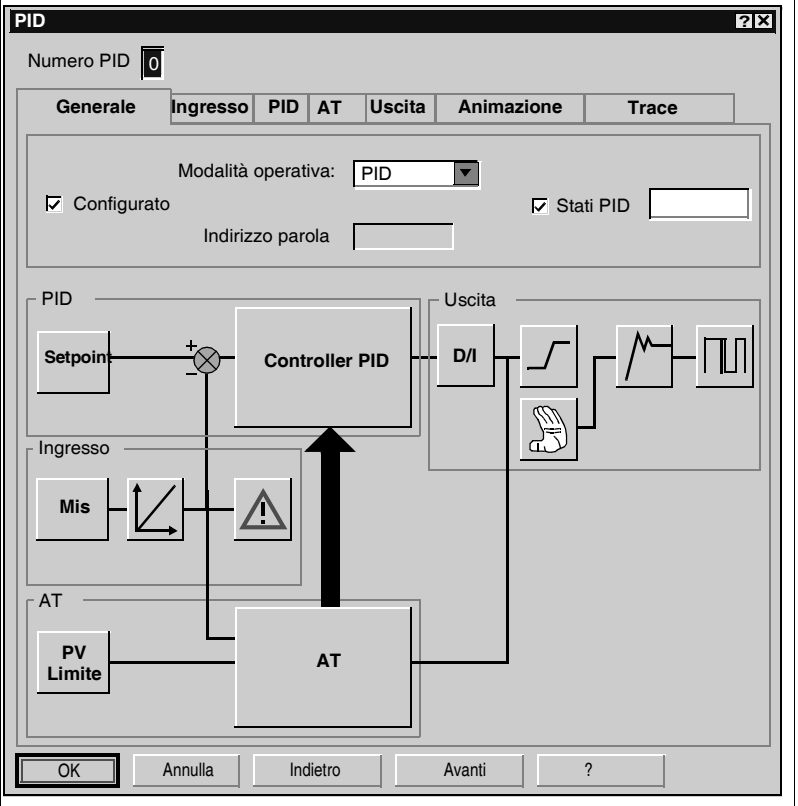
- **PID** = Regolatore PID semplice.
- **AT + PID** = La funzione Auto-Tuning verrà attivata all'avvio del PID e permetterà di specificare automaticamente i valori del guadagno **Kp, Ti, Td** (scheda **PID**) e il tipo di azione del PID (scheda **Uscita**). Alla fine della sequenza di Auto-Tuning, il regolatore passerà in modalità PID sul setpoint regolato e con i parametri regolati tramite la funzione AT.
- **AT** = La funzione Auto-Tuning verrà attivata all'avvio del PID e permetterà di specificare automaticamente i valori del guadagno **Kp, Ti, Td** (scheda **PID**) e il tipo di azione del PID (scheda **Uscita**). Alla fine della sequenza, il PID si ferma e resta in attesa. Vengono specificati i valori del guadagno **Kp, Ti, Td** (scheda **PID**) e il tipo di azione del PID (scheda **Uscita**).
- **Indirizzo parola** = La selezione del modo di funzionamento del PID può essere comandata dal programma assegnando il valore desiderato all'indirizzo parola associato alla selezione:
  - %MWxx=1: il regolatore funziona in modalità PID semplice.
  - %MWxx=2: il regolatore funziona in modalità AT + PID.
  - %MWxx=3: l regolatore funziona solo in modalità AT.

  Questo tipo di configurazione, tramite l'indirizzo parola, offre all'utente la possibilità di gestire il modo di funzionamento del regolatore PID tramite il programma applicativo, ampliando le possibilità di adattare il regolatore all'uso finale.

Apertura della finestra di dialogo del PID

La seguente tabella presenta la finestra di dialogo del PID e la procedura da seguire per accedere alle varie schede di configurazione dei parametri del PID:

Passo	Azione
1	Fare doppio clic sulla voce **PID** nel browser di configurazione situato nella parte sinistra della finestra TwidoSoft, come indicato nella seguente figura: 

Passo	Azione
2	La finestra di dialogo PID viene visualizzata in primo piano e permette di specificare i vari parametri del regolatore, come indicato nella seguente figura. In modalità Locale, questa schermata comprende diverse schede: **Generale, Ingresso, PID, AT, Uscita:**  **Importante:** completare le schede nell'ordine in cui appaiono nella finestra di dialogo del PID: Generale, Ingresso, PID, AT, quindi Uscita. **Nota:** in modalità Collegato, nella scheda verranno visualizzate due schede aggiuntive, **Animation** e **Trace** che consentono, rispettivamente, la diagnostica e la visualizzazione del funzionamento del regolatore.

Modifica dinamica dei parametri

Per modificare dinamicamente i parametri del PID (durante il funzionamento e in modalità Collegato), si consiglia di specificare degli indirizzi di memoria nei campi corrispondenti, in modo da evitare che il sistema passi in modalità non collegato ogni volta che si inseriscono modifiche immediate dei valori.

Parametrizzazione della scheda Generale

La seguente tabella indica la procedura da seguire per parametrizzare la scheda **Generale** della finestra di dialogo del PID:

Passo	Azione
1	Nella scheda **Generale**, contrassegnare la casella **Configurato** per rendere attivo il PID e per poter regolare le schede successive.
2	Nell'elenco a discesa **Modalità di funzionamento**, scegliere il tipo di funzionamento desiderato (Vedi *Modalità operative, p. 500*). **Nell'esempio:** si è scelto la modalità Indirizzo di memoria e si è specificato la parola %MW17 nel campo corrispondente. La modalità di funzionamento del PID è dunque legata al valore presente in %MW17.

Parametrizzazione della scheda Ingresso

La seguente tabella indica la procedura da seguire per parametrizzare la scheda **Ingresso** della finestra di dialogo del PID:

Passo	Azione
1	Nella scheda **Ingresso**, specificare il canale analogico che ha funzione di misura nel campo corrispondente. **Nell'esempio:** è stata scelta **%IW1.0** poiché ha la funzione di misura della temperatura.
2	Confermare gli allarmi sulla soglia alta e bassa della misura, se necessario, contrassegnando la casella e completando i campi associati. **Nota:** i valori immessi possono essere dei valori fissi (specificati nei campi associati) o dei valori modificabili (specificando nei campi corrispondenti degli indirizzi di memoria: %MWxx).

Parametrizzazione della scheda PID

La seguente tabella indica la procedura da seguire per parametrizzare la scheda **PID** della finestra di dialogo del PID:

Passo	Azione
1	Nella scheda **PID**, specificare il valore che serve a fissare il setpoint del regolatore. In generale, questo valore è un indirizzo di memoria o un setpoint emesso da un ingresso analogico. **Nell'esempio:** è stato specificato %MW0, che servirà come parola di setpoint.
2	Regolare i parametri **Kp, Ti, Td**. **Importante:** se si sceglie la modalità **AT** o **AT+PID**, è indispensabile specificare i campi Kp, Ti e Td con degli **indirizzi di memoria**, che permettono alla funzione Auto-Tuning di immettere automaticamente i valori trovati. **Nell'esempio:** è stato specificato %MW10 per Kp, %MW11 per Ti e %MW12 per Td. **Nota:** è piuttosto difficile determinare a priori i valori ottimali di regolazione di Kp, Ti e Td per un'applicazione non ancora realizzata. Di conseguenza, si consiglia di regolare in questi campi degli indirizzi di parole memoria, che permettano di regolare questi valori in modalità collegato e che evitino così il passaggio in modalità non collegato ogni volta che si eseguono modifiche immediate di valori.
3	Regolare il **Periodo di campionamento** del PID. Questo valore è quello utilizzato dal regolatore per l'acquisizione delle misure e l'aggiornamento delle uscite. **Nell'esempio:** è stato regolato il periodo di campionamento del PID a 100, ossia 1s. Con il sistema regolato con una costante di tempo di parecchi minuti, l'ordine di grandezza del periodo di campionamento sembra corretto. **Importante:** si consiglia di regolare il periodo di campionamento su un multiplo del periodo di scansione del PLC e in maniera coerente con il sistema regolato.

Parametrizzazione della scheda AT

La seguente tabella indica la procedura da seguire per parametrizzare la scheda **AT** della finestra di dialogo del PID:

Passo	Azione
1	Nella scheda **AT**, contrassegnare la casella **Autorizza** se intende utilizzare la funzione AT.
2	Specificare il valore **Limite della misura**. Si tratta del valore limite che la misura non deve superare durante la procedura diAT.
3	Specificare il valore **Setpoint di uscita**, che è il valore inviato in uscita del regolatore per generare la procedura diAT.
Nota speciale	**Per maggiori dettagli sulla regolazione di questi valori, vedere la sezione *Scheda AT della funzione PID, p. 535*.**
Suggerimento	**Si consiglia di regolare in questi campi degli indirizzi di parole memoria, che permettono di regolare questi valori in modalità collegato, evitando così il passaggio in modalità non collegato ogni volta che si eseguono modifiche immediate di valori.**

Parametrizzazione della scheda Uscita

La seguente tabella indica la procedura da seguire per parametrizzare la scheda **Uscita** della finestra di dialogo del PID:

⚠ AVVERTENZA

RISCHIO DI SOVRACCARICO DEL SISTEMA

Si ricorda che la modalità manuale agisce direttamente sull'uscita del regolatore. Di conseguenza, l'invio di un setpoint (campo **Uscita**) agisce direttamente sul sistema comandato con loop aperto. Conviene quindi gestire con attenzione questo modo di funzionamento.

La mancata osservanza di questa precauzione può causare gravi rischi per l'incolumità personale o danni alle apparecchiature.

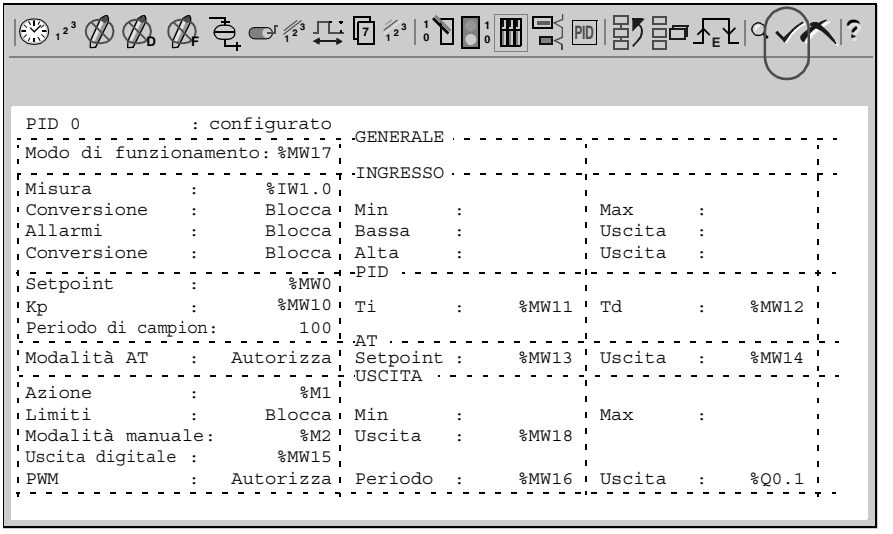
Passo	Azione
1	Nella scheda **Uscita**, specificare la selezione dell'elenco a discesa **Azione**. Questa scelta dipende dal sistema regolato: • Azione diretta: l'uscita del regolatore diminuisce quando il valore della deviazione (setpoint - misura) aumenta (regolatore freddo). • Azione inversa: l'uscita del regolatore aumenta quando il valore della deviazione (setpoint - misura) diminuisce (regolatore caldo). **Importante:** se si utilizza la funzione AT, viene automaticamente evidenziata la voce **Indirizzo bit**. La modalità di funzionamento sarà determinata dalla funzione AT e specificata nel bit associato al campo in questo caso.
2	Se necessario, regolare nel campo **Allarmi** i limiti del valore di uscita del regolatore. Questa funzione può essere necessaria su alcune applicazioni per gestire gli allarmi di processo in caso di supero delle soglie.
3	Regolare il modo di funzionamento della **modalità Manuale**. L'elenco a discesa propone varie scelte: • **Blocca** = modalità manuale non disponibile. • **Autorizza** = il regolatore funziona soltanto in modalità manuale. • **Indirizzo bit** = il valore del bit permette di modificare la modalità di funzionamento (bit a 0 = modalità automatica, bit a 1 = modalità manuale). **Nell'esempio:** è stato scelto %M2 per attivare la selezione e %MW18 e regolare manualmente il valore di setpoint.
4	Regolare la parola **Uscita digitale**. Questo modo è utilizzato dal regolatore per inviare il setpoint di regolazione. Il setpoint può essere inviato direttamente su un canale analogico di uscita (%QW..) o su una parola di memoria (%MWxx) in previsione di un'elaborazione successiva. **Importante:** se si utilizza la funzione PWM, specificare un indirizzo di memoria (%MWxx) in questo campo.

Passo	Azione
5	Se necessario, regolare l'**Uscita PWM**. **1.** Contrassegnare la casella **Autorizza** se si intende comandare il sistema tramite un attuatore PWM. **2.** Specificare il **Periodo** di comando PWM nel relativo campo. **3.** Specificare l'**Uscita** utilizzata per comandare l'attuatore PWM. Si consiglia di utilizzare le uscite transistor delle basi del controller per questa funzione (ad esempio, %Q0.0 o %Q0.1 per la base del controller TWDLMDA20DRT).
6	Convalidare la configurazione del regolatore premendo il pulsante **OK** situato in basso a sinistra dello schermo.
7	Se è necessario configurare più regolatori PID, utilizzare il pulsante **Avanti** per incrementare il numero del PID da regolare.

Editor di configurazione del PID

Dopo aver convalidato la parametrizzazione del PID, occorre convalidare l'editor di configurazione del PID, che riassume l'insieme dei parametri di ogni PID configurato.

Per convalidare la schermata di configurazione, premere l'icona **Accetta** nella barra delle icone, come indicato nella seguente figura:



Fase 4 - Inizializzazione dell'implementazione della regolazione

Prerequisiti per l'implementazione

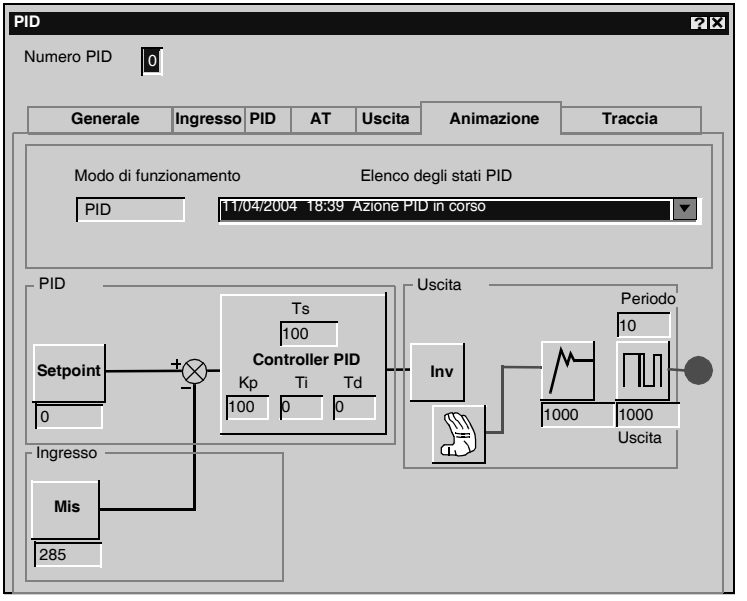
Prima dell'implementazione, rispettare le seguenti fasi:

Fase	Azione
1	Collegare il PC al controller ed effettuare il trasferimento dell'applicazione.
2	Impostare il controller è in modalità RUN.

Nota: prima di impostare il controller in RUN, verificare che le condizioni di funzionamento della macchina siano appropriate rispetto alla parte restante dell'applicazione.

Procedura L'inizializzazione dell'implementazione della regolazione, richiede i seguenti passi:

Fase	Azione
1	Creare una tabella d'animazione contenente i principali oggetti utili alla diagnostica. **Nell'esempio:** ● %MW0: setpoint del regolatore ● %IW1.0: misura ● %M0: convalida del regolatore ● %M1: tipo d'azione del regolatore (posizionato dalla funzione AT) ● %M2: scelta della modalità Automatica o Manuale ● da %MW10 a %MW12: coefficienti del regolatore PID ● %MW13: limite della misura da non superare in modalità AT ● %MW14: setpoint d'uscita del regolatore in modalità AT ● %MW15: uscita digitale del regolatore PID (specificato dal regolatore) ● %MW16: regolazione del periodo PWM ● %MW17: scelta della modalità di funzionamento del regolatore PID ● %MW18: setpoint manuale associato alla scelta del bit %M2.
2	Verificare la coerenza del valore misurato nel campo di %IW1.0. **Nell'esempio:** **1.** Si misurano 248 punti quando il sistema è stabile e a freddo. **2.** Questo valore sembra coerente perché abbiamo un coefficiente moltiplicatore di 10 tra la temperatura e il valore letto. Si può altresì, per essere sicuri, influenzare esteriormente la misura per assicurarsi che la lettura sia coerente (ossia, fare salire la temperatura vicino alla sonda per verificare la variazione della misura). **Nota:** questo test è importante in quanto il funzionamento del regolatore dipende essenzialmente dalla qualità e dalla accuratezza della misura. **3.** In caso di dubbi sull'accuratezza della misura, impostare il controller in modalità STOP e verificare i collegamenti tra gli ingressi e la scheda analogica (usare un voltmetro o un amperometro per gli ingressi 0-10V / 4-20mA, un ohmmetro per i PT100 (100 ohm à 20°) o Termocoppia (qualche decina di ohm): ● Staccare prima la sonda dai morsetti della scheda analogica. ● Verificare che i collegamenti non siano invertiti (i colori dei fili collegati agli ingressi, cavo di compensazione per i PT100). **Attenzione:** I canali degli ingressi IN0 e IN1 hanno un potenziale comune ai morsetti (-). ● Verificare che la scheda analogica sia alimentata con 24 VCC sui questi primi due morsetti. ● Verifica che i sensori d'ingresso a 4-20 mA siano alimentati. Le schede degli ingressi analogici di Twido non sono delle sorgenti di corrente.

Fase	Azione
3	Per poter avviare il regolatore, occorre iniziare a comandare il regolatore PID in modalità**Manuale** per poter valutare i valori limiti necessari alla funzione AT. Per posizionare il regolatore in modalità Manuale: **1.** Impostare il controller è in modalità RUN. **2.** Immettere nella tabella d'animazione gli indirizzi di memoria con i valori seguenti: ● %M2: Scelta della modalità manuale = 1 (M2=1 => Modalità Manuale, M2=0 => Modalità Automatica), ● %MW16: regolazione del periodo PWM = 10, ● %MW17: Scelta della modalità di funzionamento del regolatore PID = 1 (PID semplice), ● %MW18: Setpoint manuale associato alla scelta del bit %M2 = 1000. La scelta di questo valore di setpoint può effettuarsi in più volte a condizione di attendere che il sistema ritorni allo stato iniziale. **Nell'esempio:** è stato scelto il valore 1000 che corrisponde a un valore medio di aumento della temperatura (promemoria, 2000 pti = 200°). A freddo, il sistema si avvia da 250 punti sulla misura.
4	Verificare che il controller sia in modalità RUN. (%M0: convalida regolatore = 1, da specificare nella tabella d'animazione)
5	Fare doppio clic sull'elemento **PID** nel browser di configurazione.
6	Attivare la scheda **Animazione** per il numero di PID desiderato e verificare che l'animazione sia conforme alla schermata seguente:  **Nota:** le schermate del regolatore PID vengono aggiornate soltanto se il regolatore viene confermato (e il controller è in RUN).

Fase	Azione
7	Attivare la scheda **Traccia** per il numero di PID desiderato, poi: **1.** Regolare l'elenco a scorrimento del tempo su **15mn** per visualizzare la traccia della progressione del segnale di misura. **2.** Verificare che il valore di misura rientri nei valori di tolleranza del sistema. È quello che viene verificato con la scheda Traccia dove è possibile visualizzare l'incremento della misura. Quando la misura è stabilizzata, aumentare il valore corrispondente alla stabilizzazione della curva di misura (350 pti dell'esempio che corrispondono a 35°, ovvero una progressione di 10° rispetto allo stato iniziale).
8	Dall'elenco a scorrimento del tempo impostare 15mn per visualizzare la traccia della progressione del segnale di misura. Verificare che il valore della misura rientri nei valori di tolleranza del sistema: sulla scheda Traccia, si può verificare l'incremento della misura; quando la misura è stabilizzata, aumentare il valore corrispondente alla stabilizzazione della curva di misura (350 pti dell'esempio che corrispondono a 35°, ovvero una progressione di 10° rispetto allo stato iniziale).

9	Se si verifica che l'azionamento non è comandato, occorre verificare il circuito d'uscita:
	● Se l'uscita è analogica, verificare quindi la tensione o la corrente d'uscita della scheda analogica.
	● Se l'uscita è PWM, verificare:
	  ● la spia dell'uscita in questione (%Q0.1, nell'esempio)
	  ● i collegamenti delle alimentazioni e il circuito 0V per le uscite della base TWDLMDA20DRT
	  ● l'alimentazione in potenza dell'azionamento.

10	Chiudere la schermata di visualizzazione del PID e arrestare la modalità manuale posizionando nella tabella di animazione i seguenti valori: ● %M0: Convalida regolatore = 0 (Arresto del regolatore) ● %M2: Scelta della modalità Automatica o Manuale = 0 (Arresto della modalità Manuale) ● %MW17: Scelta della modalità di funzionamento del regolatore PID = 0 ● %MW18: Setpoint manuale associato alla scelta del bit %M2 = 0

Fase 5 - Implementazione della regolazione AT + PID

Introduzione

Questa sezione descrive la parametrizzazione del regolatore per l'avvio del funzionamento in modalità AT+PID. In questa modalità di funzionamento, il regolatore esegue automaticamente la regolazione del regolatore sui coefficienti Kp, Ti, Td.

Nota: per non compromettere la regolazione finale, si consiglia di evitare ogni possibile interferenza esterna con il sistema durante la regolazione. Allo stesso modo, prima di avviare la sequenza AT è opportuno assicurarsi che il sistema sia stabilizzato.

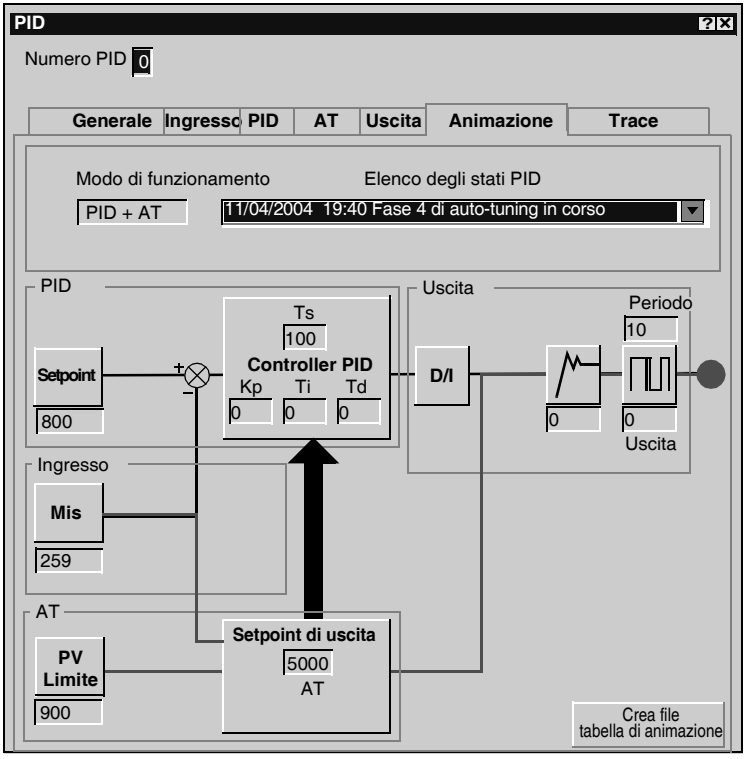
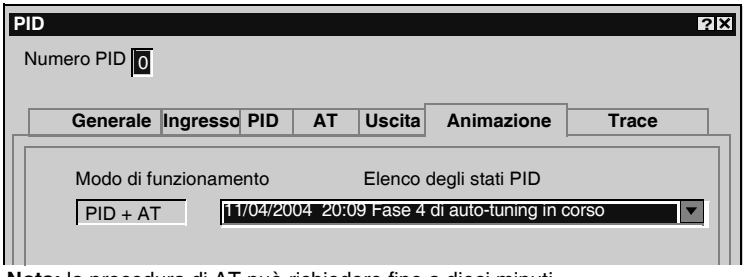
Note sulla parametrizzazione di Kp, Ti, Td

Per permettere il funzionamento in modalità AT+PID, è necessario che siano soddisfatte le seguenti condizioni:

- La parametrizzazione dei coefficienti **Kp, Ti, Td** deve essere regolata su degli **indirizzi di memoria (%MWxx)**.
- La parametrizzazione del tipo di **Azione** nella **scheda Uscita** deve essere regolata su un **indirizzo di memoria bit (%Mxx)**.

Per posizionare il regolatore in modalità AT+PID, procedere come segue:

Passo	Azione
1	Specificare o verificare l'indirizzo di memoria con i seguenti valori nella tabella d'animazione: • %M2: scelta della modalità Automatica o manuale = 0, • %MW0: setpoint regolatore = 600 (nell'esempio, il setpoint sarà attivato dopo la sequenza AT e il regolatore garantirà una temperatura di 60°), • da %MW10 a %MW12: coefficienti del regolatore PID (lasciare a 0, la sequenza AT immetterà i valori adeguati), • %MW13: limite della misura da non superare in modalità AT = 900 (nell'esempio, non superare 90°, altrimenti si verificherà un errore AT), • %MW14: setpoint di uscita del regolatore in modalità AT = 2000 (generato dalla prova in modalità manuale). Si tratta del valore della modifica della fase applicata al processo. In modalità AT, il setpoint di uscita è applicato direttamente all'uscita del regolatore. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. Il valore di questa parola deve essere dunque compreso tra 0 e 10 000. **Nota:** il valore del setpoint di uscita della funzione di auto-tuning deve sempre essere superiore al valore dell'ultima uscita applicata al processo. • %MW15: uscita digitale del regolatore PID (specificata dal regolatore), • %MW16: regolazione del periodo PWM (lasciare il valore 10 regolato in precedenza), • %MW17: scelta della modalità di funzionamento del regolatore PID = 2 (AT + PID), • %MW18: setpoint manuale associato alla scelta del bit %M2 = 0.
2	Configurare il controller Twido in modo che esegua una scansione in **modalità periodica.**
3	Regolare la **Durata** del periodo di scansione del controller Twido in modo che il valore del **Periodo di campionamento (Ts)** del regolatore PID ne sia un multiplo esatto. **Nota:** per ulteriori informazioni sulla definizione del periodo di campionamento, vedere le sezioni *Requisiti della sintonizzazione automatica, p. 553* e *Metodi per la determinazione del periodo di campionamento (Ts), p. 554*.
4	Verificare che il controller sia in modalità RUN.
5	Immettere il bit di memoria %M0. %M0: convalida regolatore = 1 nella tabella d'animazione.
6	Fare doppio clic sulla voce **PID** nel browser di configurazione.

Passo	Azione
7	Attivare la scheda **Animazione** per il numero di PID desiderato e verificare che l'animazione sia conforme alla seguente schermata:  **Nota:** le schermate del regolatore PID vengono aggiornate soltanto se è stato convalidato il regolatore (e se il controller è in RUN).
8	fare clic sulla scheda **Trace** e attendere che il sistema avvii la sequenza di AT.  **Nota:** la procedura di AT può richiedere fino a dieci minuti.

Memorizzazione dei coefficienti calcolati Kp, Ti, Td

Al termine della sequenza diAuto-Tuning, le parole di memoria assegnate ai coefficienti Kp, Ti, Td sono completate con i valori calcolati. Questi valori vengono scritti nella memoria RAM e salvati nel controller a condizione che l'applicazione sia valida (messa fuori tensione inferiore a 30 giorni) e che non si verifichi un riavvio a freddo (%S0).

Nota: se il sistema non è in condizioni di instabilità, a causa di variazioni esterne, i valori possono essere scritti in maniera **definitiva** nella parametrizzazione del regolatore PID e il regolatore può passare solo in modalità PID.

Ripetizione della sequenza AT

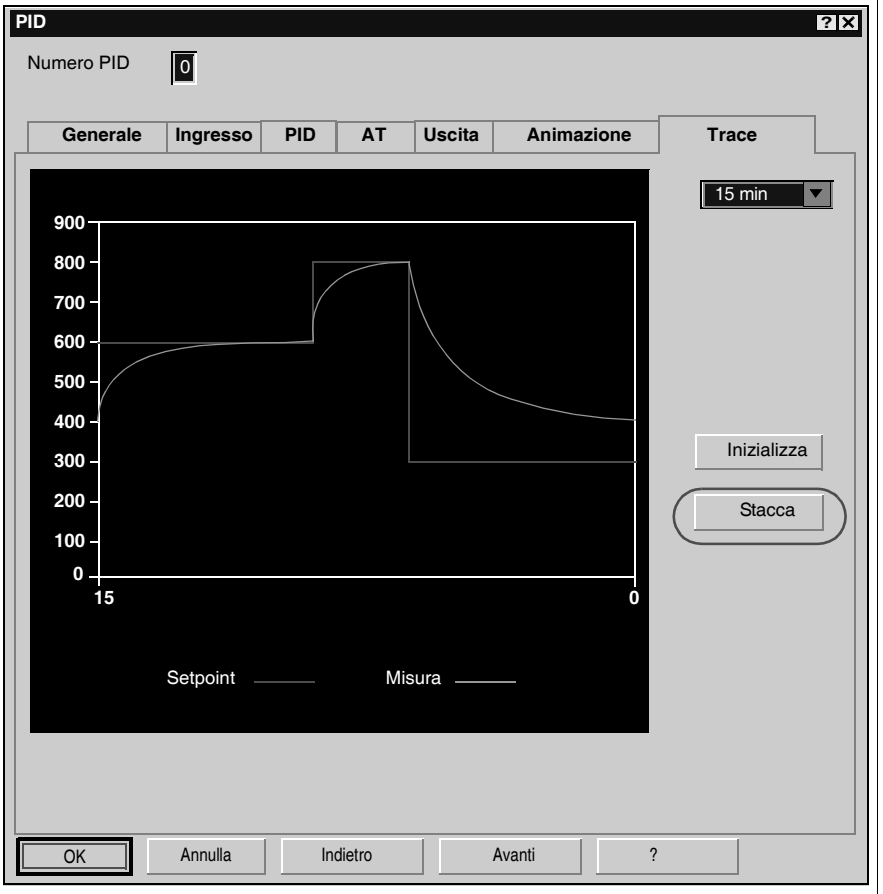
La sequenza di Auto-Tuning viene ripetuta ogni volta che si mette il sistema in RUN o che si esegue un riavvio a freddo (%S0).
Si consiglia inoltre di testare le parole di diagnostica tramite il programma per determinare la procedura da seguire in caso di riavvio.

Fase 6 - Debug delle regolazioni

Accesso alla tabella di animazione

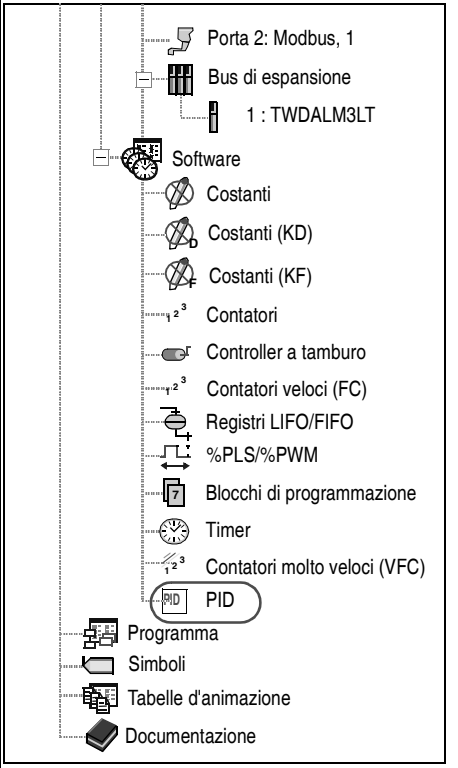
Per facilitare il debug del sistema, l'accesso alla tabella di animazione è sempre possibile quando le schermate del regolatore PID si trovano in primo piano.

Nota:
Nel caso in cui la visualizzazione delle curve del setpoint e della misura sia possibile soltanto con il pulsante **Stacca** della scheda **Trace** (vedere la finestra della scheda Trace che segue), si può accedere alla tabella di animazione tramite il menu **Finestra** → **Editor delle tabelle di animazione - Animazione...**



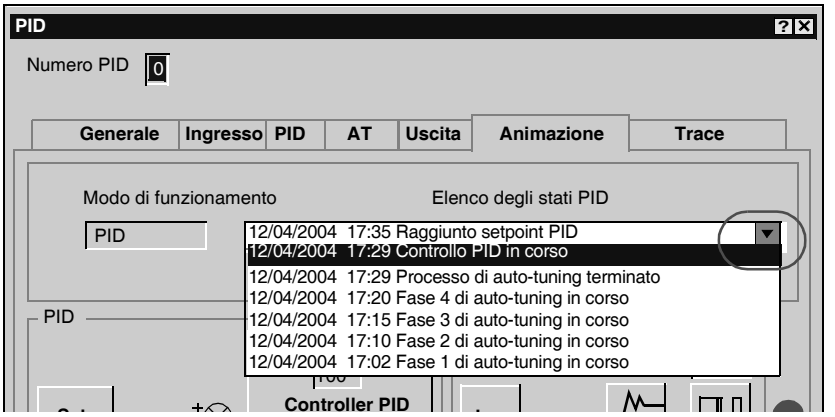
Ritorno alle schermate PID

Per ritornare alle schermate del regolatore PID senza perdere la cronologia del tracciato delle curve, procedere come segue:

Passo	Azione
1	Fare doppio clic sulla voce **PID** nel browser situato nella parte sinistra della schermata TwidoSoft (vedere la finestra del browser riportata nella seguente illustrazione): 
2	Quando compare la finestra del regolatore PID, selezionare il numero del PID desiderato nella scheda **Generale**.

Cronologia degli stati PID

Nella scheda **Animazione** dei regolatori PID, facendo clic sull'elenco a discesa, si può accedere agli ultimi 15 stati del regolatore in uso, come indicato nella seguente figura:



Nota: gli stati del PID vengono memorizzati quando il PC e TwidoSoft sono in modalità Collegato al controller.

17.4 Funzione PID

In breve

Oggetto di questa sezione

Questa sezione descrive il comportamento, le funzionalità e l'implementazione della funzione PID.

Nota: per ulteriori informazioni sulla configurazione rapida del controller PID e della sintonizzazione automatica PID, consultare la *Guida all'avvio rapido della funzione PID Twido, p. 493*.

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Panoramica	519
Principio del loop di regolazione	520
Metodologia di sviluppo di un'applicazione di regolazione	521
Compatibilità e prestazioni	522
Caratteristiche dettagliate della funzione PID	523
Accesso alla configurazione del PID	526
Scheda Generale del PID	528
Scheda Ingresso del PID	531
Scheda PID del PID	533
Scheda AT della funzione PID	535
Scheda Uscita del PID	540
Accesso al debug del PID	543
Scheda Animazione del PID	545
Scheda Traccia del PID	547
Stati PID e codici di errore	549
Sintonizzazione del PID con sintonizzazione automatica (Auto-Tuning, AT)	553
Metodo di regolazione dei parametri PID	561
Ruolo e influenza dei parametri di un PID	564
Appendice 1: Fondamenti della teoria PID	568
Appendice 2: Modello di prim'ordine con ritardo	570

Panoramica

Generale

La funzione di regolazione PID è una funzione del linguaggio di programmazione TwidoSoft.
Consente di programmare loop di regolazione PID su controller compatibili con TwidoSoft versione 2.0 o successiva.

Questa funzione è particolarmente adatta per:
- rispondere alle necessità dei processi sequenziali che richiedono funzioni di regolazione ausiliarie (esempi: macchine per imballaggi a pellicola plastica, macchine per trattamento delle superfici, presse),
- rispondere alle necessità dei processi di regolazione semplice (esempi: forni per il trattamento dei metalli, forni per ceramiche, gruppi frigoriferi di piccole dimensioni).

L'implementazione è molto semplice e si basa su schermate di:
- Configurazione
- e di debug,

insieme a una linea di programma (blocco Operate in linguaggio Ladder o semplice chiamata del PID in linguaggio List) che indica il numero di PID utilizzato.
Esempio di linea di programma in linguaggio Ladder:



Nota: in qualsiasi applicazione di automazione Twido, il numero massimo di funzioni PID configurabili è 14.

Funzionalità principali

Le funzionalità principali sono le seguenti:
- ingresso analogico,
- conversione lineare della misura configurabile,
- allarme alto e basso nell'ingresso configurabile,
- uscita analogica o PWM,
- limitazione dell'uscita configurabile,
- azione diretta o inversa configurabile.

Principio del loop di regolazione

In breve

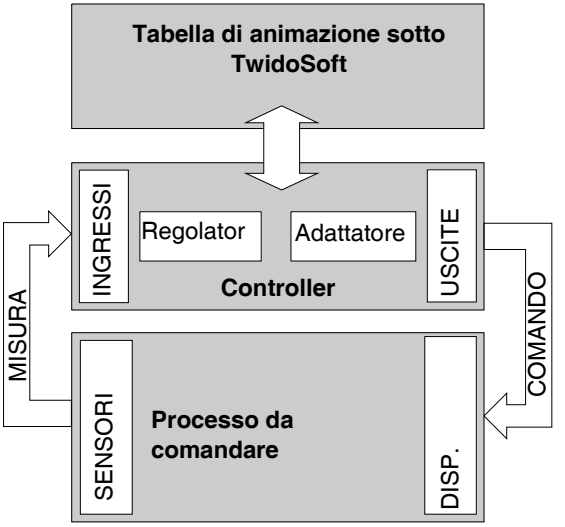
Il funzionamento di un loop di regolazione comprende tre fasi distinte:
- l'acquisizione dei dati:
  - misure che provengono dai sensori del processo (analogici, encoder),
  - setpoint provenienti in genere da variabili interne del controller o da dati originati da una tabella di animazione TwidoSoft.
- l'esecuzione dell'algoritmo di regolazione PID,
- l'emissione dei comandi adatti alle caratteristiche dei dispositivi di azionamento da pilotare tramite uscite digitali (PWM) o analogiche.

L'algoritmo PID elabora il segnale da comandare a partire:
- dalla misura campionata dal modulo di ingresso,
- dal valore del setpoint fissato dall'operatore o dal programma,
- dai valori dei diversi parametri del correttore.

Il segnale proveniente dal correttore viene elaborato direttamente da una scheda di uscita analogica del controller collegato al dispositivo di comando oppure elaborato attraverso un adattamento PWM su un'uscita digitale del controller.

Illustrazione

L'illustrazione seguente mostra lo schema del principio di un loop di regolazione.

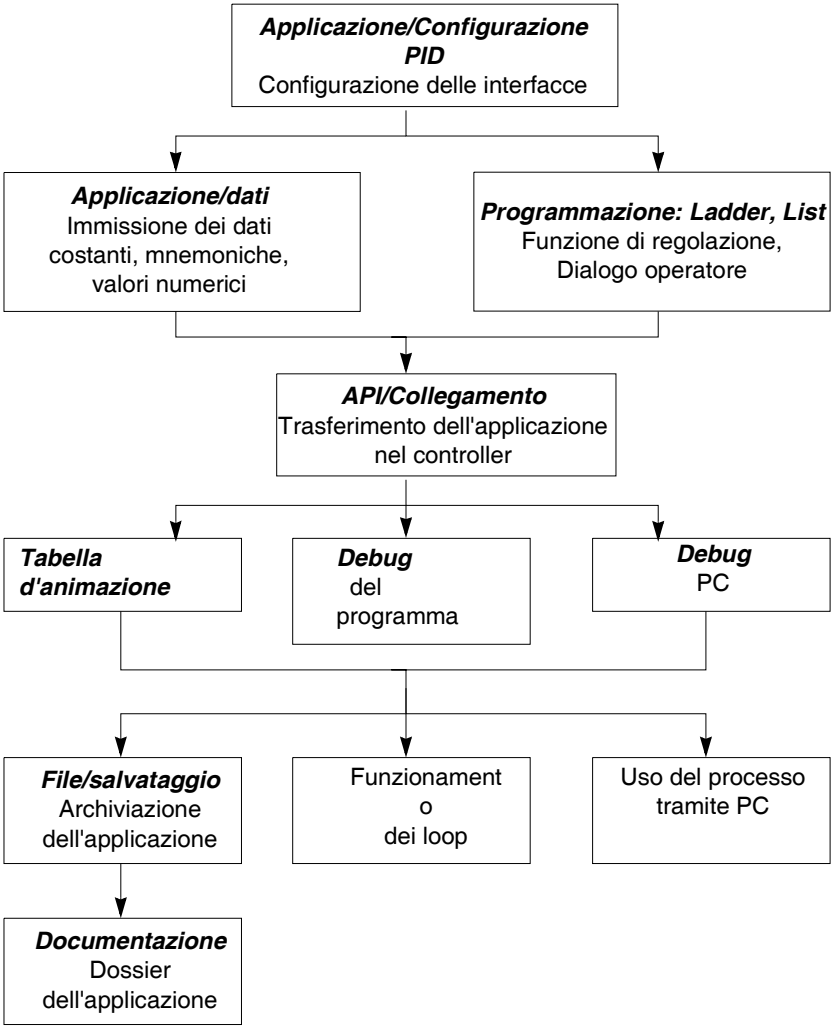


Metodologia di sviluppo di un'applicazione di regolazione

Principio di funzionamento

Lo schema seguente presenta l'insieme dei task da effettuare al momento della creazione e del debug di un'applicazione di regolazione.

Nota: l'ordine definito dipende dal metodo di lavoro adottato dall'utente. La sequenza proposta ha solo carattere indicativo.



Compatibilità e prestazioni

In breve

La funzione PID di Twido è una funzionalità disponibile per controller compatibili con TwidoSoft versione 2.0 o successive. Per questo motivo la sua installazione dipende dalle compatibilità hardware e software descritte nei paragrafi seguenti.
Questa funzionalità, inoltre, richiede le risorse presentate nel paragrafo **Prestazioni**.

Compatibilità

La funzione PID di Twido è disponibile sui controller Twido con versione software 2.0 o successiva.
Se si dispone di versioni precedenti di Twido, è possibile aggiornare il firmware per poter utilizzare la funzione PID.

Nota: i moduli di ingresso e uscita analogici della versione 1.0 possono essere utilizzati in ingressi o uscite del PID senza aggiornamento.

Per poter configurare e programmare un PID su queste versioni hardware diverse, deve essere disponibile la versione **2.0 o successiva del software TwidoSoft**.

Prestazioni

Le prestazioni dei loop di regolazione PID sono indicate di seguito.

Descrizione	Durata
Tempo di esecuzione di un loop	0,4 ms

Caratteristiche dettagliate della funzione PID

Generalità

La funzione PID realizza una correzione PID a partire da una misura e da un setpoint analogico nel formato predefinito [0 - 10000] e fornisce un comando analogico nello stesso formato o una modulazione d'impulso su un'uscita digitale (PWM).
Tutti i parametri del PID sono descritti nelle finestre che ne consentono la configurazione. Di seguito verranno indicate in breve le funzioni disponibili con le relative misure e verrà descritta la loro integrazione al PID in un diagramma di flusso funzionale.

Nota: per un utilizzo a scala piena (risoluzione ottimale), è possibile configurare l'ingresso analogico collegato alla sezione di misura del PID in 0-10000. Tuttavia, se si utilizza la configurazione predefinita (0-4095), il controller funziona correttamente.

Nota: per un corretto funzionamento della regolazione, il controller Twido deve **obbligatoriamente essere in modalità periodica**. La funzione PID viene quindi eseguita periodicamente a ogni ciclo e il campionamento dei dati di ingresso del PID rispetterà il periodo fissato durante la configurazione (vedere tabella di seguito).

Informazioni sulle funzioni disponibili

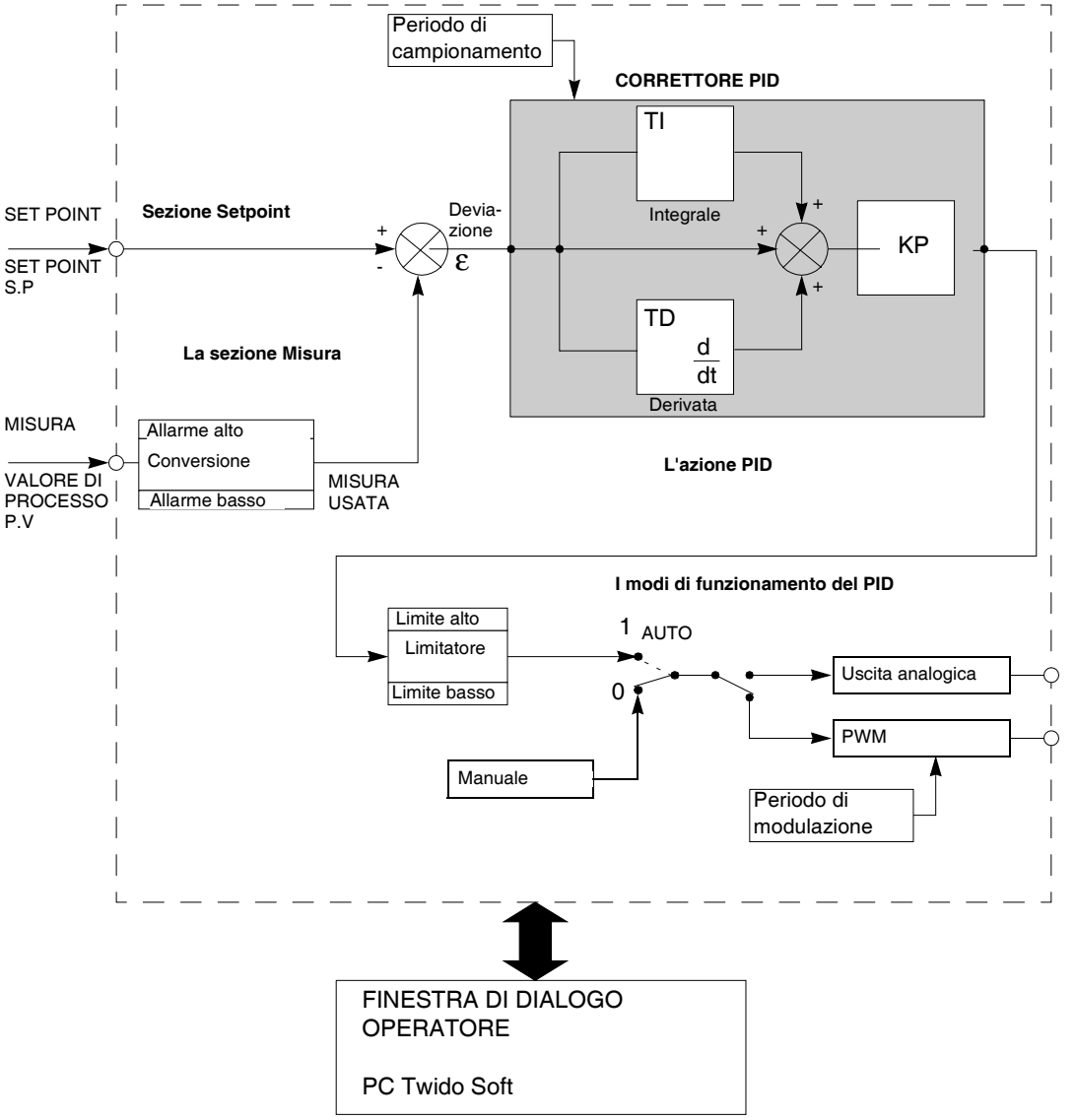
La tabella seguente indica le diversi funzioni disponibili e le relative scale.

Funzione	Scala e commento
Conversione lineare dell'ingresso	La conversione consente di passare da un valore compreso tra 0 e 10000 (risoluzione modulo di ingresso analogico) a un valore compreso tra -32768 e 32767
Guadagno proporzionale	Associato a un fattore di 100, il valore è compreso tra 1 e 10000, corrispondente a un guadagno variabile tra 0,01 e 100. **Nota:** se si immette un valore non valido per il guadagno (guadagno negativo o nullo), TwidoSoft ignora questa impostazione dell'utente e assegna automaticamente il valore predefinito 100 a questo fattore.
Tempo integrale	Associato0 a una base tempo di 0,1 secondi, il valore è compreso tra 0 e 20000, che corrisponde a un tempo integrale compreso tra 0 e 2000,0 secondi.
Tempo derivato	Associato0 a una base tempo di 0,1 secondi, il valore è compreso tra 0 e 10000, corrispondente a un tempo derivato compreso tra 0 e 1000,0 secondi.
Periodo di campionamento	Associato a una base tempo di 0,01 secondi, il valore è compreso tra 1 e 10000, corrispondente a un periodo compreso tra 0,01 e 100 secondi.
Uscita PWM	Associata a una base tempo di 0,1 secondi, il valore è compreso tra 1 e 500, corrispondente a un periodo di modulazione compreso tra 0,1 e 50 secondi.
Uscita analogica	Valore compreso tra 0 e +10000.
Allarme alto per variabile di processo	Questo allarme viene impostato dopo la conversione. È compreso tra -32768 e 32767 se la conversione è attivata e tra 0 e 10000 in caso contrario.
Allarme basso per variabile di processo	Questo allarme viene impostato dopo la conversione. È compreso tra -32768 e 32767 se la conversione è attivata e tra 0 e 10000 in caso contrario.
Limite alto per l'uscita	Questo valore limite è compreso tra 0 e 10000 per un valore di uscita analogica. Quando la modulazione di ampiezza degli impulsi è attiva, il limite corrisponde a una percentuale del periodo modulato. 0% per 0 e 100% per 10000.
Limite basso per l'uscita	Questo valore limite è compreso tra 0 e 10000 per un valore di uscita analogica. Quando la modulazione di ampiezza degli impulsi è attiva, il limite corrisponde a una percentuale del periodo modulato. 0% per 0 e 100% per 10000.
Modalità manuale	Se è attivata la modalità manuale, l'uscita è uguale a un valore fisso definito dall'utente. Questa uscita è compresa tra 0 e 10000 (da 0 a 100% per un'uscita PWM).
Azione diretta o inversa	L'azione diretta o inversa è disponibile e agisce direttamente sull'uscita.
Sintonizzazione automatica (Auto-Tuning, AT)	Questa funzione sintonizza automaticamente i parametri Kp, Ti, Td e di azione diretta/inversa per ottenere la convergenza ottimale del processo di controllo.

Nota: per una migliore comprensione dell'azione di ciascuna delle funzioni descritte nella tabella precedente, consultare lo schema seguente.

Principio di funzionamento

Lo schema che segue presenta il principio di funzionamento della funzione PID.



Nota:la descrizione dei parametri utilizzati è presentata nella tabella della pagina precedente e nelle schermate di configurazione.

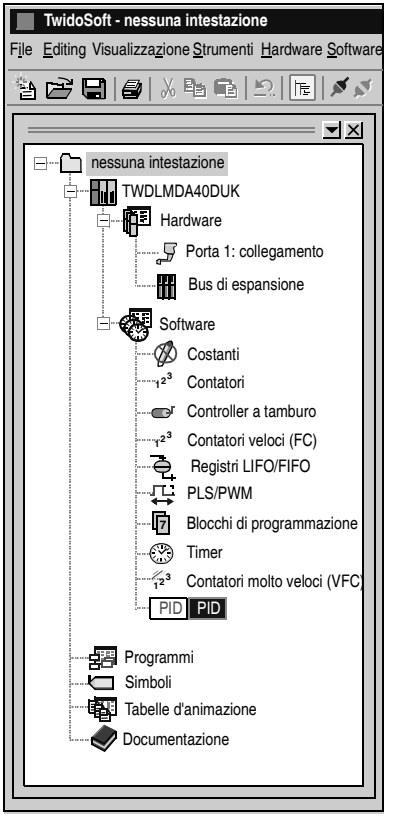
Accesso alla configurazione del PID

In breve

Di seguito viene descritto come accedere alle schermate di configurazione di un PID sui controller TWIDO.

Procedura

La tabella che segue descrive la procedura per accedere alle schermate di configurazione di un PID :

Passo	Azione
1	Verificare di essere in modalità **locale**.
2	Aprire il browser. **Risultato:** 

Passo	Azione
3	Fare doppio clic su **PID**. **Risultato:** viene visualizzata la finestra di parametrizzazione dei PID, posizionata in modo predefinito sulla scheda **Generale** (Vedi *Scheda Generale del PID, p. 528*). **Nota**: è anche possibile fare clic con il pulsante destro del mouse su **PID** e scegliere l'opzione **Modifica** oppure selezionare il menu **Software** → **PID**o utilizzare il menu **Programma** → **Editor di configurazione** → **Icona PID**. In quest'ultimo caso è anche possibile scegliere PID e fare clic sull'icona **lente d'ingrandimento** per selezionare un PID specifico.

Scheda Generale del PID

In breve

Quando si apre il PID dal browser, viene visualizzata la finestre di configurazione del PID. In questa finestra è possibile eseguire le seguenti operazioni:

- configurazione di ogni PID TWIDO,
- debug di ogni PID TWIDO.

Quando si visualizza questa schermata, può essere attiva:

- la modalità Locale: viene visualizzata in modo predefinito la scheda **Generale** e si ha accesso ai parametri di configurazione,
- a modalità Collegato: viene visualizzata la scheda **Animazione** e si ha accesso ai parametri di debug e di regolazione.

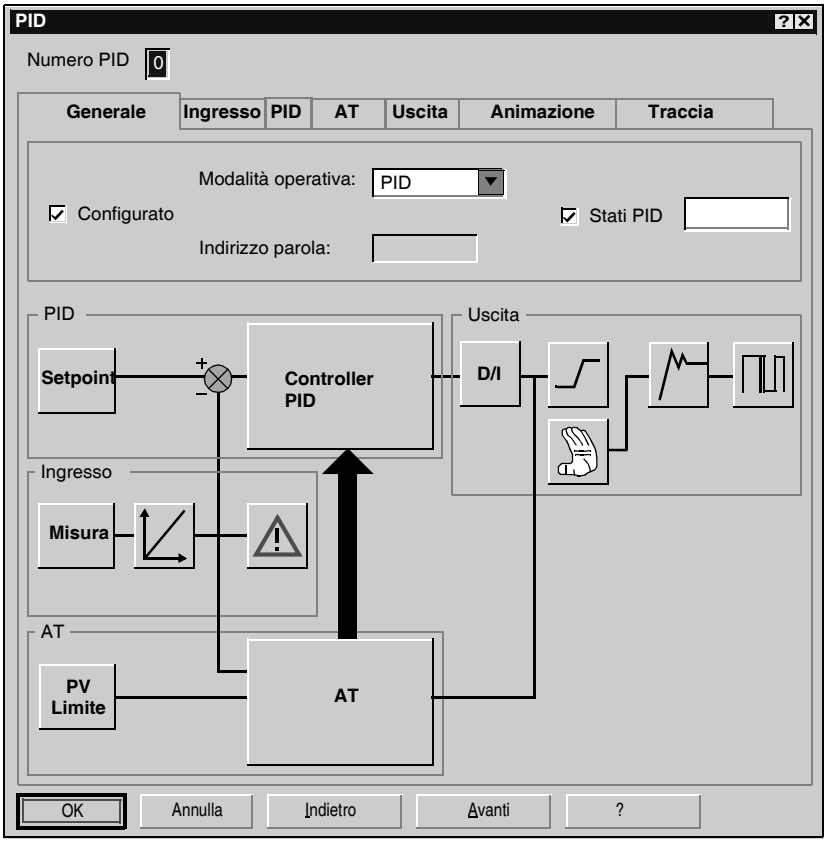
Nota: in alcuni casi le schede e i campi non sono accessibili per le ragioni seguenti:
È selezionata la modalità operativa "Solo PID", che impedisce l'accesso ai parametri della scheda AT non più necessari.

- La modalità operativa (offline o online) attualmente attiva non consente di accedere a questi parametri.
- È selezionata la modalità operativa "Solo PID", che impedisce l'accesso ai parametri della scheda AT non più necessari.

Di seguito è riportata la descrizione della scheda **Generale**.

Scheda Generale del PID

La schermata seguente consente di specificare i parametri generali del PID.



Descrizione

Nella tabella che segue vengono descritti i parametri che è possibile definire.

Campo	Descrizione
Numero PID	Indicare qui il numero del PID da configurare. Il valore è compreso tra 0 e 13, 14 PID al massimo per applicazione.
Configurato	Per configurare il PID, questa casella deve essere selezionata, altrimenti non sarà possibile eseguire alcuna azione su queste schermate né utilizzare il PID, anche se presente nell'applicazione.
Modalità operativa	Indicare la modalità operativa desiderata. Sono disponibili tre modalità operative e un indirizzo di parola: ● PID ● AT ● AT+PID ● Indirizzo parola
Indirizzo parola	In questa casella di testo si può immettere una parola interna (da %MW0 a %MW2999) che permette di impostare la modalità operativa con la programmazione. La parola interna può assumere tre valori a seconda della modalità operativa che si desidera impostare: ● %MWx = 1 (per impostare solo PID) ● %MWx = 2 (per impostare AT + PID) ● %MWx = 3 (per impostare solo AT)
Stati PID	Se si seleziona questa casella per attivare l'opzione, nella casella di testo si può immettere una parola di memoria (da %MW0 a %MW2999) che viene usata dal controller PID per memorizzare lo stato PID corrente durante l'esecuzione del controller PID e/o della funzione di sintonizzazione automatica (per maggiori informazioni, vedere *Stati PID e codici di errore, p. 549*.)
Diagramma	Questo diagramma consente di visualizzare le diverse possibilità offerte per la configurazione del PID.

Scheda Ingresso del PID

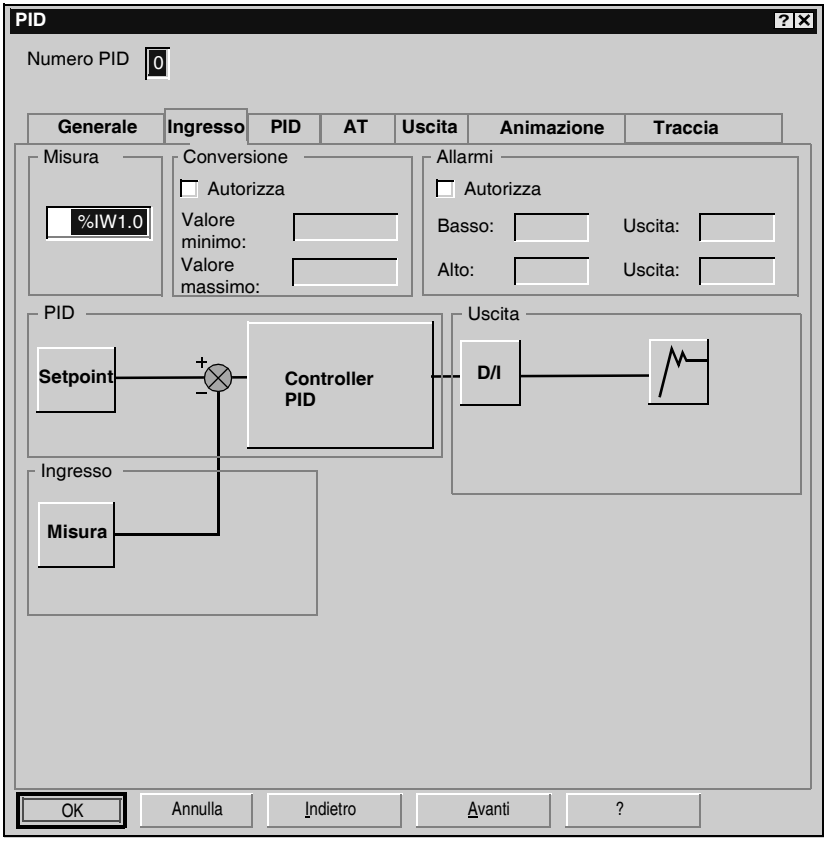
In breve

Questa scheda consente di specificare i parametri di ingresso del PID.

Nota: la scheda è accessibile in modalità locale.

Scheda Ingresso del PID

La schermata seguente consente di specificare i parametri degli ingressi del PID.



Descrizione Nella tabella che segue vengono descritti i parametri che è possibile definire.

Campo	Descrizione
Numero PID	Indicare qui il numero del PID da configurare. Il valore è compreso tra 0 e 13, 14 PID al massimo per applicazione.
Misura	Indicare la variabile che conterrà il valore della misura del processo da gestire. La scala predefinita è da 0 a 10000. È possibile immettere una parola interna (da %MW0 a %MW2999) oppure un ingresso analogico (da %IWx.0 a %IWx.1).
Conversione	Selezionare questa casella per eseguire una conversione della variabile di processo specificata come ingresso del PID. Se si seleziona questa casella, i due campi **Valore minimo** e **Valore massimo** sono accessibili. La conversione è lineare e converte un valore compreso tra 0 e 10,000 in un valore il cui minimo e massimo è compreso tra -32768 e +32767.
Valore minimo **Valore massimo**	Indicare i valori minimo e massimo della scala di conversione. La variabile di processo viene riscalata automaticamente nell'intervallo **[da valore Min a valore Max]**. **Nota**: il **Valore minimo** deve essere obbligatoriamente inferiore al **Valore massimo**. **Valore minimo** o **Valore massimo** possono essere parole interne (da %MW0 a %MW2999), costanti interne (da %KW0 a %KW255) oppure un valore compreso tra -32768 e +32767.
Allarmi	Selezionare questa casella per attivare allarmi su variabili di ingresso. **Nota**: i valori di allarme devono essere determinati rispetto alla variabile di processo ottenuta dopo la fase di conversione. Devono quindi essere compresi tra il **Valore minimo** e il **Valore massimo** quando la conversione è attiva. Altrimenti saranno compresi tra **0 e 10000.**
Basso **Uscita**	Indicare il valore dell'allarme basso nel campo **Basso**. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. **Uscita** deve contenere l'indirizzo del bit che verrà azzerato in caso sia raggiunto il limite basso. **Uscita** può essere un bit interno (da %M0 a %M255) oppure un'uscita (da %Qx.0 a %Qx.32).
Alto **Uscita**	Indicare il valore dell'allarme alto nel campo **Alto**. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. **Uscita** deve contenere l'indirizzo del bit che verrà impostato a 1 quando viene raggiunto il limite massimo. **Uscita** può essere un bit interno (da %M0 a %M255) oppure un'uscita (da %Qx.0 a %Qx.32).
Diagramma	Questo diagramma consente di visualizzare le diverse possibilità offerte per la configurazione del PID.

Scheda PID del PID

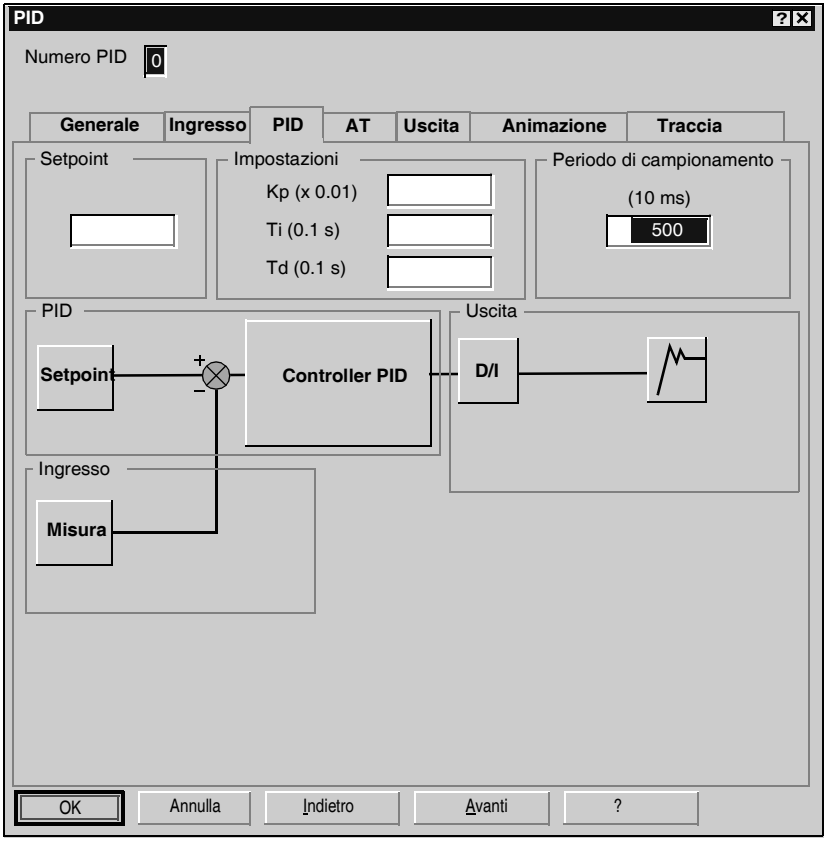
In breve

Questa scheda consente di specificare i parametri interni del PID.

Nota: la scheda è accessibile in modalità locale.

Scheda PID del PID

La schermata seguente consente di specificare i parametri interni del PID.



Descrizione Nella tabella che segue vengono descritti i parametri che è possibile definire.

Campo	Descrizione
Numero PID	Indicare qui il numero del PID da configurare. Il valore è compreso tra 0 e 13, 14 PID al massimo per applicazione.
Setpoint	Indicare qui il valore del setpoint PID. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. Questo valore deve quindi essere compreso tra 0 e 10000 quando la conversione è inibita. In caso contrario deve essere compreso tra il valore **Min** e il valore **Max** per la conversione.
Kp * 100	Indicare qui il coefficiente proporzionale del PID moltiplicato per 100. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. L'intervallo valido per il parametro Kp è: 0 < Kp < 10000. **Nota:** Se Kp viene erroneamente impostato a 0 (Kp ≤ 0 non è valido), il valore predefinito Kp=100 viene assegnato automaticamente dalla funzione PID.
TI (0,1 sec)	Indicare qui il coefficiente d'azione integrale in rapporto a una base tempo di 0,1 secondi. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. Il valore deve essere compreso tra 0 e 20000. **Nota:** per disattivare l'azione integrale del PID, impostare questo coefficiente a 0.
Td (0,1 sec)	Indicare qui il coefficiente d'azione derivata in rapporto a una base tempo di 0,1 secondi. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. Il valore deve essere compreso tra 0 e 10000. **Nota:** per disattivare l'azione derivativa del PID, impostare questo coefficiente a 0.
Periodo di campionamento	Indicare qui il periodo di campionamento del PID per una base tempo di 10^{-2} secondi (10 ms). Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. Il valore deve essere compreso tra 1 (0.01 s) e 10000 (100 s).
Diagramma	Questo diagramma consente di visualizzare le diverse possibilità offerte per la configurazione del PID.

Nota: quando è attivata la funzione AT, i parametri Kp, Ti e Td non sono più impostati dall'utente ma configurati automaticamente e programmaticamente dall'algoritmo AT. In questo caso, occorre immettere in questi campi solo una **parola interna** (da %MW0 a %MW2999).
Attenzione: non immettere una costante interna o un valore diretto quando la funzione AT è attivata, perché in questo modo si provoca un errore durante l'esecuzione dell'applicazione PID.

Scheda AT della funzione PID

In breve

L'impostazione dei parametri PID corretti può essere noiosa, lunga e difficoltosa. Per gli utenti non professionisti, benché esperti, la configurazione del controllo di processo quindi può risultare difficile, così come la sintonizzazione ottimale.
L'algoritmo di sintonizzazione automatica PID è concepito per determinare automaticamente e correttamente i seguenti quattro termini PID:

- fattore di guadagno,
- valore integrale,
- valore derivato e
- azione diretta o inversa.

La funzione AT è pertanto in grado di fornire una sintonizzazione rapida e ottimale del ciclo di processo.

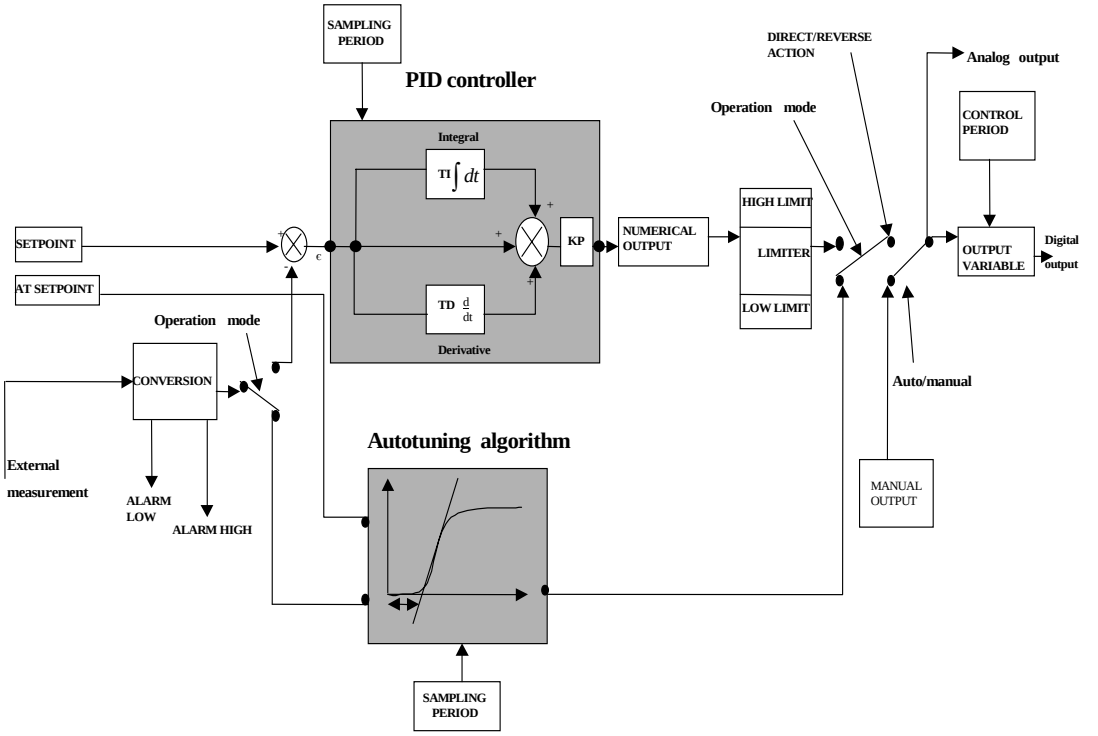
Requisiti della funzione AT

La sintonizzazione automatica PID è particolarmente adatta per i processi di controllo della temperatura.
In linea generale, i processi controllabili con la funzione AT devono soddisfare i requisiti seguenti:

- il processo deve essere per la maggior parte lineare sull'intero campo operativo,
- la risposta di processo a una modifica di livello dell'uscita analogica deve seguire un modello asintotico dei transienti, e
- vi deve essere poco disturbo nelle variabili di processo. In caso di processo di controllo della temperatura, questo implica che non deve sussistere una percentuale anormalmente elevata di scambio di calore tra il processo e l'ambiente.

Principio di funzionamento della funzione AT

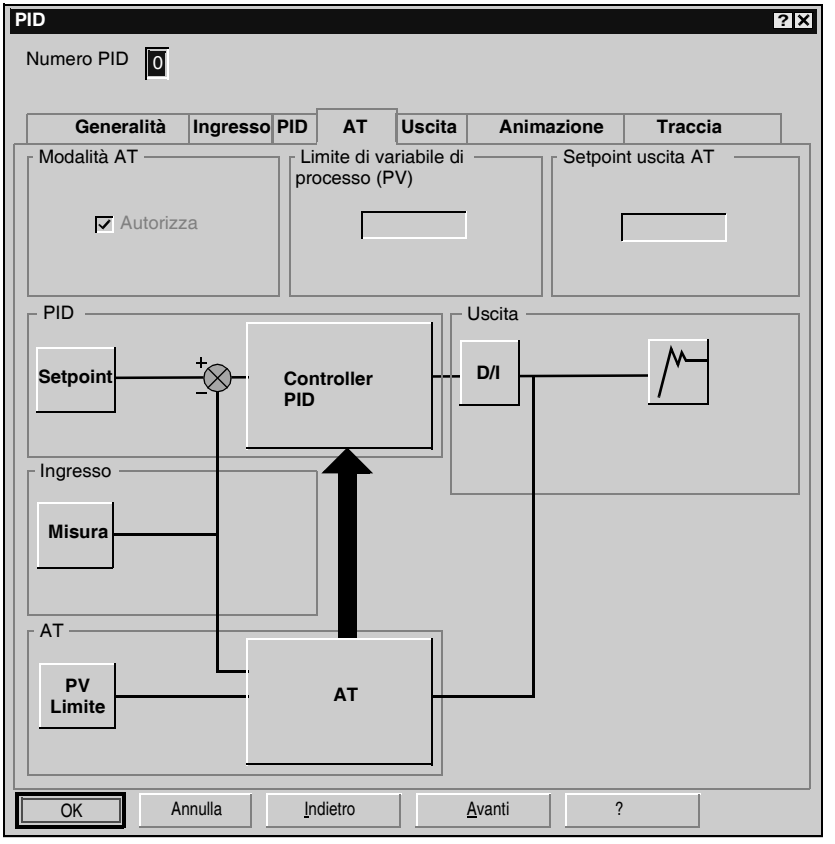
Lo schema che segue descrive il principio di funzionamento della funzione PID e le modalità di interazione con i cicli PID:



Scheda AT della funzione PID

La schermata seguente consente di attivare/disattivare la funzione AT e di immettere i parametri AT.

Nota: la schermata è accessibile solo in modalità offline.



Descrizione

⚠ AVVERTENZA

IL LIMITE DELLA VARIABILE DI PROCESSO (PV) E I VALORI DEI SETPOINT DI USCITA DEVONO ESSERE IMPOSTATI CON MOLTA ATTENZIONE.

La funzione di sintonizzazione automatica PID è un processo a ciclo aperto che agisce direttamente sul controllo di processo senza regolazione o limitazione tranne quelle del limite della variabile di processo (PV) e dei setpoint di uscita. Entrambi questi valori devono quindi essere selezionati con attenzione nell'ambito dell'intervallo consentito dal processo in modo da evitare possibili sovraccarichi del processo.

La mancata osservanza di questa precauzione può causare gravi rischi per l'incolumità personale o danni alle apparecchiature.

Nella tabella che segue vengono descritti i parametri che è possibile definire.

Campo	Descrizione
Autorizza	Selezionare questa casella per attivare la modalità AT. Esistono due modi per selezionare questa casella, a seconda se si sta impostando la modalità operativa manualmente o tramite un indirizzo di parola nella scheda Generale della funzione PID: ● Se si imposta la **modalità operativa** su **PID+AT** o **AT** nella scheda **Generale** (vedere *Scheda Generale del PID, p. 528*), l'opzione Autorizza viene selezionata automaticamente ed è impossibile deselezionarla. ● Se si imposta la modalità operativa tramite un indirizzo di parola %MWx (%MWx = 2: PID+AT; %MWx = 3: AT), è necessario selezionare l'opzione Autorizza manualmente per consentire la configurazione dei parametri AT. **Risultato:** in entrambi i casi descritti sopra tutti i campi della schermata di configurazione della scheda AT diventano attivi e occorre immettere i valori appropriati nei campi Setpoint e Uscita.
Limite di variabile di processo (PV)	Specificare il limite che la variabile di processo misurata non deve superare durante il processo AT. Questo parametro fornisce sicurezza al sistema di controllo, dato che la funzione AT è un processo a ciclo aperto. Questo valore può essere una parola interna (da %MW0 a un massimo di %MW2999, a seconda della quantità di memoria di sistema disponibile), una costante interna (da %KW0 a %KW255) o un valore diretto. Questo valore deve quindi essere compreso tra 0 e 10000 quando la conversione è inibita. In caso contrario deve essere compreso tra il valore Min e il valore Max per la conversione.
Setpoint uscita AT	Indicare qui il valore di uscita AT. Questo è il valore della variazione a gradino applicata al processo. Questo valore può essere una parola interna (da %MW0 a %MW2999), una costante interna (da %KW0 a %KW255) oppure un valore diretto. Questo valore deve quindi essere compreso tra 0 e 10000 quando la conversione è inibita. **Nota:**il setpoint di uscita AT deve essere sempre maggiore dell'ultima uscita applicata al processo.

Nota: quando la funzione AT è attivata, le costanti (%KWx) e i valori diretti non sono più consentiti. Nei seguenti insiemi di campi PID sono consentite solo parole di memoria:

- I parametri **Kp, Ti** e **Td** devono essere impostati come **parole di memoria** (%MWx) nella scheda PID;
- Il campo **Azione** viene impostato automaticamente su **"Bit indirizzo"** nella scheda OUT;
- La casella **Bit** deve essere riempita con un **bit di memoria** (%Mx) adeguato nella scheda OUT.

Coefficienti Kp, Ti, Td calcolati

Una volta terminato il processo AT, i coefficienti PID Kp, Ti e Td calcolati:

- vengono memorizzati nelle rispettive parole di memoria (%MWx) e
- possono essere visualizzati nella scheda **Animazione**, solo con TwidoSoft in modalità online.

Scheda Uscita del PID

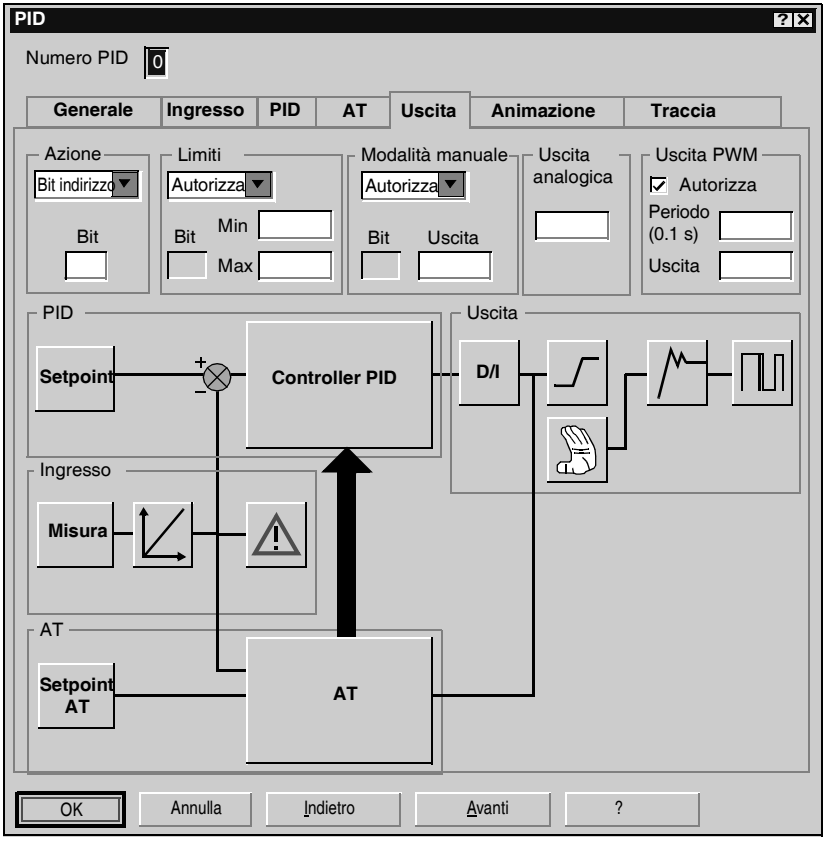
In breve

Questa scheda consente di specificare i parametri di uscita del PID.

Nota: la scheda è accessibile in modalità locale.

Scheda Uscita del PID

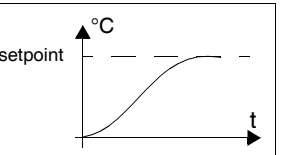
La schermata seguente consente di specificare i parametri interni del PID.



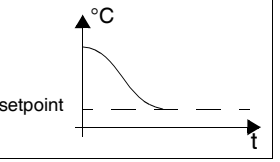
Descrizione Nella tabella che segue vengono descritti i parametri che è possibile definire.

Campo	Descrizione
Numero PID	Indicare qui il numero del PID da configurare. Il valore è compreso tra 0 e 13, 14 PID al massimo per applicazione.
Azione	Specificare qui il tipo di azione PID sul processo. Sono disponibili tre opzioni: **Inversa**, **Diretta** o **Indirizzo bit**. Se è stato selezionato **indirizzo bit,** è possibile modificare questo tipo da programma, modificando il bit associato che è un bit interno (da %M0 a %M255) o un ingresso (da %Ix.0 a %Ix.32). L'azione è diretta se il bit è a 1, altrimenti è inversa. **Nota:** quando è attivata la funzione AT, l'algoritmo di sintonizzazione automatica determina automaticamente il tipo corretto di azione diretta o inversa per il processo di controllo. In questo caso, è disponibile solo un'opzione nell'elenco a discesa Azione: **Bit indirizzo.** Nella casella di testo associata **Bit** occorre immettere una **parola interna** (da %MW0 a %MW2999). Non tentare di immettere una costante interna o un valore diretto nella casella di testo **Bit**, perché in questo modo si provoca un errore di esecuzione.
Limiti Bit	Indicare qui se si desidera limitare l'uscita del PID. Sono disponibili tre opzioni: **Autorizza**, **Disattiva** o **Indirizzo bit**. Se è stato selezionato **indirizzo bit,** è possibile attivare (bit a 1) o disattivare (bit a 0) la gestione dei limiti da programma, modificando il bit associato che è un bit interno (da %M0 a %M255) o un ingresso (da %Ix.0 a %Ix.32).
Min **Max**	Impostare qui i limiti massimo e minimo per l'uscita PID. **Nota**: **Min** deve essere obbligatoriamente inferiore a **Max**. **Min** o **Max** possono essere parole interne (da %MW0 a %MW2999), costanti interne (da %KW0 a %KW255) oppure un valore compreso tra 1 e 10000.
Modalità manuale Bit Uscita	Indicare qui se si desidera impostare la modalità manuale per il PID. Sono disponibili tre opzioni: **Autorizza**, **Disattiva** o **Indirizzo bit**. Se è stato selezionato **Indirizzo bit,** è possibile passare alla modalità manuale (bit a 1) o alla modalità automatica (bit a 0) da programma, modificando il bit associato che è un bit interno (da %M0 a %M255) o un ingresso (da %Ix.0 a %Ix.32). L'**Uscita** della modalità manuale deve contenere il valore da assegnare all'uscita analogica quando il PID è in modalità manuale. Tale **Uscita** può essere una parola (da %MW0 a %MW2999) o un valore diretto nel formato [0-10000].
Uscita analogica	Indicare qui l'uscita del PID in modalità automatica. Questa **Uscita analogica** può essere di tipo %MW (da %MW0 a %MW2999) o %QW (%QWx.0).
Uscita PWM autorizzata Periodo (0,1s) Uscita	Se si desidera utilizzare la funzione PWM del PID, selezionare la casella. Specificare il periodo di modulazione in **Periodo (0.1s)**. Questo periodo deve essere compreso tra 1 e 500 e può essere una parola interna (da %MW0 a %MW2999) o una costante interna (da %KW0 a %KW255). Specificare il bit dell'uscita PWM come il valore in **Uscita**. Può essere un bit interno (da %M0 a %M255) oppure un'uscita (da %Qx.0 a %Qx.32).
Diagramma	Questo diagramma consente di visualizzare le diverse possibilità offerte per la configurazione del PID.

Nota: il termine Inversa nel campo Azione viene utilizzato per raggiungere un setpoint alto (es.: per il riscaldamento)



il termine Diretta nel campo Azione viene utilizzato per raggiungere un setpoint basso (es.: per il raffreddamento)



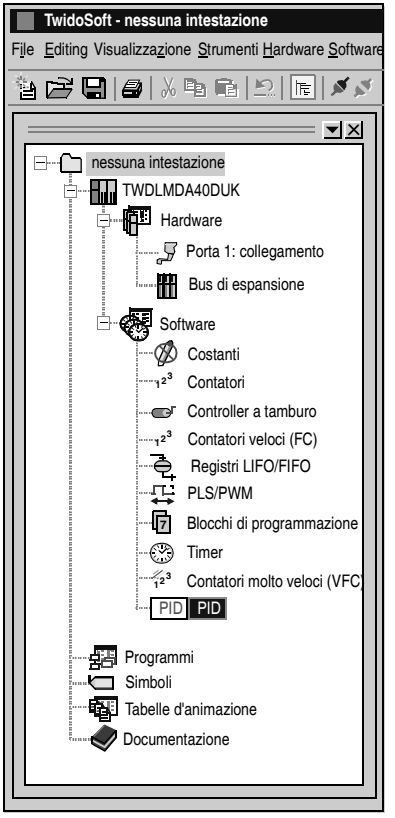
Accesso al debug del PID

In breve

Di seguito viene descritto come accedere alle schermate di debug di un PID sui controller TWIDO.

Procedura

La tabella che segue descrive la procedura per accedere alle schermate di debug di un PID :

Passo	Azione
1	Verificare di essere in modalità **Collegato**.
2	Aprire il browser. **Risultato:** 

Passo	Azione
3	Fare doppio clic su **PID**. **Risultato:** viene visualizzata la finestra di parametrizzazione dei PID, posizionata in modo predefinito sulla scheda **Animazione** (Vedi *Scheda Animazione del PID, p. 545*). **Nota**: è anche possibile fare clic con il pulsante destro del mouse su **PID** e scegliere l'opzione **Modifica** oppure selezionare il menu **Software** → **PID**o utilizzare il menu **Programma** → **Editor di configurazione** → **Icona PID**. In quest'ultimo caso è anche possibile scegliere PID e fare clic sull'icona **lente d'ingrandimento** per selezionare un PID specifico.

Scheda Animazione del PID

In breve

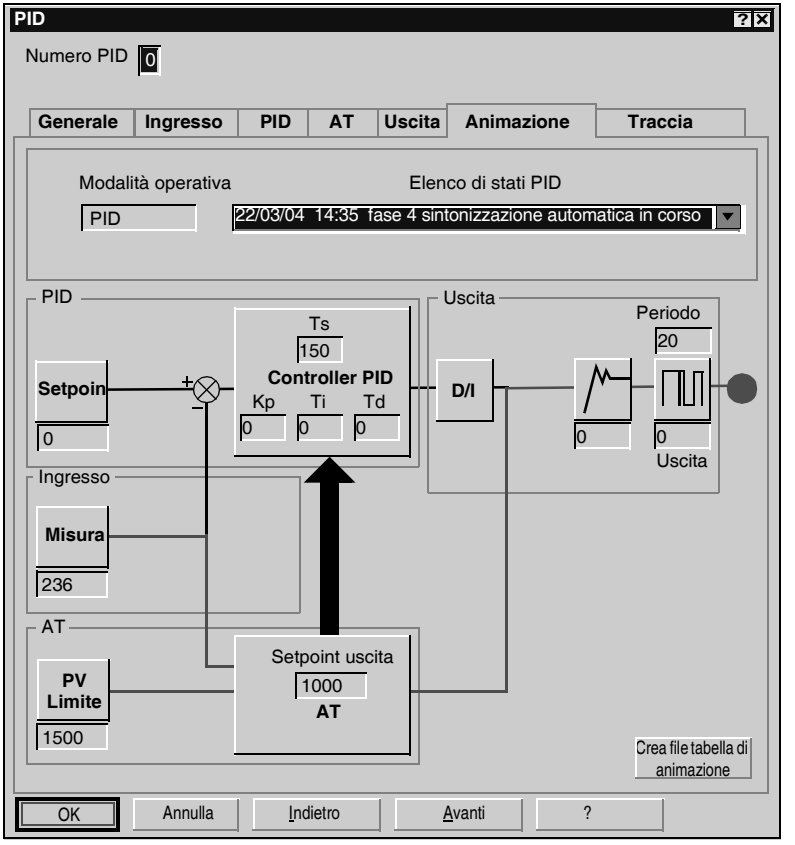
Questa scheda consente di eseguire il debug del PID.
Il diagramma visualizzato dipende dal tipo di PID creato. Vengono infatti visualizzati solo gli elementi configurati.

La visualizzazione è dinamica. I collegamenti attivi sono presentati in rosso e quelli non attivi in nero.

Nota: la scheda è accessibile in modalità online.

Scheda Animazione del PID

La schermata seguente consente di e di eseguire il debug del PID.



Descrizione La tabella che segue descrive le varie aree della finestra:

Campo	Descrizione
Numero PID	Indicare qui il numero del PID di cui eseguire il debug. Il valore è compreso tra 0 e 13, 14 PID al massimo per applicazione.
Modalità operativa	Questo campo visualizza la modalità operativa corrente del PID.
Elenco di stati PID	L'elenco a discesa permette di visualizzare in tempo reale gli ultimi 15 stati del PID. Ogni variazione di stato aggiorna l'elenco indicando la data e l'ora e lo stato corrente.
Crea un file tabella di animazione	Fare clic sul pulsante **Crea un file tabella di animazione** per creare un file contenente tutte le variabili visualizzate nel diagramma, in modo da poter eseguire le modifiche in linea ed eseguire il debug del PID

Scheda Traccia del PID

In breve

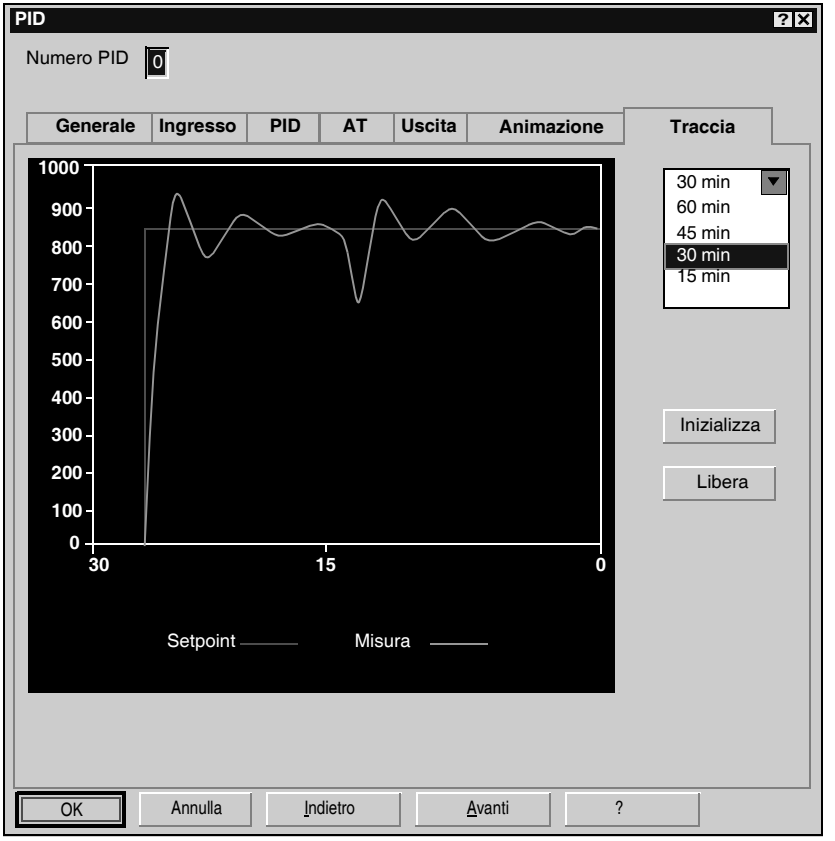
Questa scheda consente di visualizzare il funzionamento del PID e impostarne quindi il comportamento.

I grafici inizano ad essere tracciati non appena viene visualizzata la finestra di debug.

Nota: la scheda è accessibile in modalità online.

Scheda Animazione del PID

La schermata seguente consente di visualizzare l'impostazione del PID.



Descrizione La tabella che segue descrive le varie aree della finestra:

Campo	Descrizione
Numero PID	Indicare qui il numero del PID da visualizzare. Il valore è compreso tra 0 e 13, 14 PID al massimo per applicazione.
Grafico	Quest'area visualizza i trend del **setpoint** e del **valore di processo**. La scala dell'asse orizzontale (X) è determinata dal menu situato in alto a destra. La scala dell'asse verticale è determinata dai valori di configurazione dell'ingresso del PID (con o senza conversione). Viene ottimizzata in modo automatico in modo da visualizzare al meglio i trend.
Menu della scala dell'asse orizzontale	Questo menu consente di modificare la scala orizzontale. È possibile scegliere tra quattro valori: 15, 30, 45 o 60 minuti.
Inizializza	Questo pulsante cancella il trend e riavvia la visualizzazione dei tracciati.

Stati PID e codici di errore

In breve

Oltre all'**Elenco di stati PID** disponibile nella finestra di dialogo **Animazione** (vedere *Scheda Animazione del PID, p. 545*) che consente di visualizzare e tornare agli ultimi 15 stati PID, il controller Twido PID è in grado di registrare lo stato corrente sia del controller PID che del processo AT in una parola di memoria definita dall'utente.
Per informazioni su come abilitare e configurare la **parola di memoria dello stato PID** (%MWi), vedere *Scheda Generale del PID, p. 528*.

Parola di memoria dello stato PID

La parola di memoria dello stato PID può registrare tre tipi di informazioni PID:
- Stato corrente del controller PID (stato PID)
- Stato corrente del processo di sintonizzazione automatica (stato AT)
- Codici di errore PID e AT

Nota: La parola di memoria dello stato PID è di sola lettura.

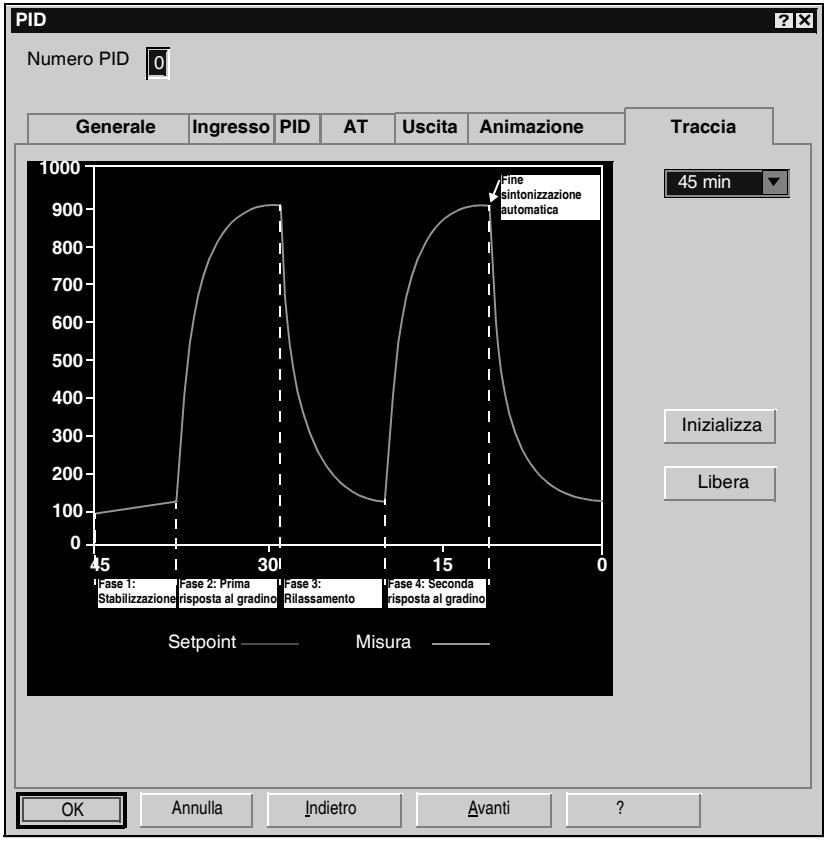
Parola di memoria dello stato PID

Di seguito viene riportata la tabella di concordanza della codifica esadecimale di stato del controller PID e parola di memoria:

Formato esadecimale dello stato PID	Descrizione
0000h	Controllo PID non attivo
2000h	Controllo PID in corso
4000h	Setpoint PID raggiunto

Descrizione dello stato AT

Il processo di sintonizzazione automatica è diviso in 4 fasi consecutive. Perché la sintonizzazione automatica possa essere terminata correttamente, deve essere eseguita ogni fase del processo. La curva e la tabella seguenti di risposta del processo descrivono le 4 fasi della sintonizzazione automatica PID del controller Twido:



Le fasi di sintonizzazione automatica vengono descritte nella tabella seguente:

Fase AT	Descrizione
1	La **fase 1** è la fase di stabilizzazione. Inizia quando l'utente avvia il processo AT. Durante questa fase, la sintonizzazione automatica Twido verifica che la variabile di processo sia in stato costante. **Nota:**l'ultima uscita applicata al processo prima dell'inizio della sintonizzazione automatica viene usata sia come punto iniziale che come punto di rilassamento per il processo di sintonizzazione automatica.
2	La **fase 2** applica la prima variazione di gradino al processo. Produce una risposta del processo simile a quella mostrata nella figura precedente.
3	La **fase 3** è la fase di rilassamento che inizia quando si è stabilizzata la prima risposta al gradino. **Nota:**il rilassamento avviene verso l'equilibrio che è determinato come l'ultima uscita applicata al processo prima dell'inizio della sintonizzazione automatica.
4	La **fase 4** applica la seconda variazione a gradino al processo per un valore e in un modo identici a quanto descritto per la fase 2. Al termine della fase 4, il processo di sintonizzazione automatica si conclude e i parametri AT vengono calcolati e memorizzati nelle rispettive parole di memoria. **Nota:**una volta terminata questa fase, la variabile di processo viene ripristinata all'ultimo livello di uscita applicato al processo prima dell'inizio della sintonizzazione automatica.

Parola di memoria dello stato AT

Di seguito viene riportata la tabella di concordanza della codifica esadecimale di stato del controller PID e parola di memoria:

Formato esadecimale dello stato AT	Descrizione
0100h	Fase 1 sintonizzazione automatica in corso
0200h	Fase 2 sintonizzazione automatica in corso
0400h	Fase 3 sintonizzazione automatica in corso
0800h	Fase 4 sintonizzazione automatica in corso
1000h	Processo di sintonizzazione automatica completo

Codici di errore PID e AT

La tabella seguente descrive i potenziali errori di esecuzione che possono verificarsi durante i processi di controllo PID e di sintonizzazione automatica:

Processi PID/AT	Codice di errore (esadecimale)	Descrizione
Errore PID	8001h	Valore della modalità operativa fuori intervallo
	8002h	Conversione lineare min e max uguale a
	8003h	Limite superiore per uscita digitale minore del limite inferiore
	8004h	Limite di variabile di processo fuori intervallo di conversione lineare
	8005h	Limite di variabile di processo minore di 0 o maggiore di 10000
	8006h	Setpoint fuori intervallo di conversione lineare
	8007h	Setpoint minore di 0 o maggiore di 10000
	8008h	Azione di controllo diversa dall'azione determinata all'avvio di AT
Errore di sintonizzazione automatica	8009h	Errore di sintonizzazione automatica: raggiunto limite di variabile di processo (PV)
	800Ah	Errore di sintonizzazione automatica: campionamento eccessivo o setpoint di uscita impostato troppo basso
	800Bh	Errore di sintonizzazione automatica: Kp è zero
	800Ch	Errore di sintonizzazione automatica: costante di tempo negativa
	800Dh	Errore di sintonizzazione automatica: ritardo negativo
	800Eh	Errore di sintonizzazione automatica: errore nel calcolo di Kp
	800Fh	Errore di sintonizzazione automatica: costante di tempo superiore a rapporto di ritardo > 20
	8010h	Errore di sintonizzazione automatica: costante di tempo superiore a rapporto di ritardo < 2
	8011h	Errore di sintonizzazione automatica: limite per Kp superato
	8012h	Errore di sintonizzazione automatica: limite per Ti superato
	8013h	Errore di sintonizzazione automatica: limite per Td superato

Sintonizzazione del PID con sintonizzazione automatica (Auto-Tuning, AT)

Panoramica della sintonizzazione del PID

La funzione di controllo PID si basa sui seguenti tre parametri definiti dall'utente: Kp, Ti e Td. La sintonizzazione del PID mira a determinare precisamente questi parametri di processo allo scopo di fornire un controllo ottimale del processo stesso.

Ambito della sintonizzazione automatica

La funzione AT del controller Twido è particolarmente adatta per la sintonizzazione automatica di processi termici. Dato che i valori dei parametri PID possono variare considerevolmente da un processo di controllo all'altro, la funzione AT del controller Twido permette di determinare con uno sforzo minore valori più precisi di quelli ottenuti considerando le ipotesi più probabili.

Requisiti della sintonizzazione automatica

Quando si usa la funzione AT, occorre accertarsi che il processo di controllo e il controller Twido soddisfino i seguenti quattro requisiti:

- Il processo di controllo deve essere un sistema stabile a loop aperto.
- All'inizio della sintonizzazione automatica, il processo di controllo deve essere in stato costante con un segnale d'ingresso del processo pari a zero (ad es. un forno o una fornace deve essere a temperatura ambiente).
- Durante la sintonizzazione automatica, accertarsi che nessun disturbo penetri nel processo in quanto ciò potrebbe falsare i parametri di calcolo o impedire la corretta esecuzione della sintonizzazione automatica (ad es. la porta del forno non deve venire aperta, neppure temporaneamente).
- Configurare il controller Twido per la scansione in **modalità periodica.** Dopo aver determinato il periodo di campionamento corretto (Ts) per la sintonizzazione automatica, occorre configurare il periodo di scansione in modo che il periodo di campionamento (Ts) sia un multiplo esatto del periodo di scansione del controller Twido.

Nota: per garantire l'esecuzione corretta del controllo PID e della sintonizzazione automatica, è fondamentale che il controller Twido venga configurato per eseguire scansioni in modalità periodica (non ciclica). In modalità periodica ogni scansione del controller inizia a intervalli di tempo regolari. In questo modo la percentuale di campionamento è costante per tutta la durata della misura (mentre in modalità ciclica una scansione inizia appena finisce quella precedente, il che rende il periodo di campionamento squilibrato da una scansione all'altra).

Modalità operative AT

La sintonizzazione automatica può essere usata indipendentemente (modalità AT) o insieme al controllo PID (AT + PID):

- **Modalità AT:** dopo la convergenza del processo AT e l'esecuzione corretta con la determinazione dei parametri di controllo PID Kp, Ti e Td (o dopo il rilevamento di un errore nell'algoritmo AT), l'uscita numerica AT è impostata a 0 e viene visualizzato il messaggio seguente nell'elenco a discesa **Elenco di stati PID** : "Sintonizzazione automatica completa".
- **Modalità AT + PID:** Innanzitutto viene eseguita la funzione AT. Dopo che la funzione AT è stata eseguita correttamente, inizia il loop di controllo PID (basato sui parametri Kp, TI e Td calcolati dalla funzione AT).
  Nota relativa a AT+PID: se si verifica un errore nell'algoritmo AT:
  - nessun parametro PID viene calcolato;
  - l'uscita numerica AT è impostata all'ultima uscita applicata al processo prima dell'inizio della sintonizzazione automatica;
  - viene visualizzato un messaggio di errore nell'elenco a discesa Elenco di stati PID;
  - il controllo PID viene annullato.

Nota: Transizione senza scatti
In **modalità AT+PID**, la transizione da AT a PID è senza scatti.

Metodi per la determinazione del periodo di campionamento (Ts)

Come verrà spiegato nelle due sezioni seguenti (vedere *Appendice 1: Fondamenti della teoria PID, p. 568* e *Appendice 2: Modello di prim'ordine con ritardo, p. 570*), il **periodo di campionamento (Ts)** è un parametro chiave del controllo PID. Il periodo di campionamento può essere ricavato dalla **costante di tempo AT (τ).**
I due metodi disponibili per valutare il periodo di campionamento corretto (Ts) usando la sintonizzazione automatica vengono descritti nelle sezioni seguenti.

- Metodo della curva di risposta del processo
- Metodo della ricerca per tentativi

Entrambi i metodi sono descritti nelle sottosezioni seguenti.

Introduzione al metodo della curva di risposta del processo

Questo metodo prevede l'impostazione di una variazione a gradino all'ingresso del processo di controllo e la registrazione della curva di uscita del processo con indicazione del tempo.
Il metodo della curva di risposta del processo parte dalla seguente ipotesi:

- Il processo di controllo può essere descritto adeguatamente come un modello di prim'ordine con ritardo mediante la seguente funzione di trasferimento:

$$\frac{S}{U} = \frac{k}{1 + \tau p} \cdot e^{-\theta p}$$

Per maggiori dettagli vedere l'Appendice 2: Modello di prim'ordine con ritardo.

Uso del metodo della curva di risposta del processo

Per determinare il periodo di campionamento (Ts) usando il metodo della curva di risposta del processo, procedere nel seguente modo:

Passo	Azione
1	Si presuppone che siano state già configurate le varie impostazioni nelle schede PID Generale, Ingresso, PID, AT e Uscita della funzione PID.
2	Selezionare la scheda **PID > Uscita** dal browser dell'applicazione.
3	Selezionare **Autorizza** o **Bit indirizzo** nell'elenco a discesa **Modalità manuale** per attivare l'uscita manuale e impostare il campo **Uscita** a un livello elevato (nell'intervallo [5000-10000]).
4	Selezionare **Controller > Trasferimento PC => Controller...** dalla barra dei menu per scaricare il programma dell'applicazione nel controller Twido.
5	Nella finestra di configurazione PID, passare alla modalità **Traccia**.
6	Eseguire la funzione PID e controllare l'aumento della curva di risposta.
7	Quando la curva di risposta ha raggiunto uno stato costante, interrompere la misura PID. **Nota:** mantenere attiva la finestra Traccia PID.
8	Usare il metodo grafico seguente per determinare la costante di tempo (τ) del processo di controllo: **1.** Calcolare l'uscita delle variabili di processo all'aumento del 63% ($S_{[63\%]}$) mediante la formula seguente: $S_{[63\%]} = S_{[iniziale]} + (S_{[finale]}-S_{[iniziale]})x63\%$ **2.** Trovare graficamente l'ascissa del tempo ($t_{[63\%]}$) che corrisponde a S(63%). **3.** Trovare graficamente il tempo iniziale ($t_{[iniziale]}$) che corrisponde all'inizio dell'aumento della risposta di processo. **4.** Calcolare la costante di tempo (τ) del processo di controllo usando la relazione seguente: $\tau = t_{[63\%]}-t_{[iniziale]}$
9	Calcolare il periodo di campionamento (Ts) in base al valore di (τ) determinato nel passo precedente, usando la regola seguente: $Ts = \tau/75$ **Nota:** l'unità di base per il periodo di campionamento è 10 ms. Occorre pertanto arrotondare in eccesso o in difetto il valore di Ts ai 10 ms più prossimi.
10	Selezionare **Programma > Modifica modalità scansione** e procedere come segue: **1.** Impostare la **Modalità di scansione** del controller Twido su **Periodica**. **2.** Impostare il **Periodo di scansione** in modo che il periodo di campionamento (Ts) sia un **multiplo esatto** del periodo di scansione, usando la regola seguente: Periodo di scansione = Ts / n, dove "n" è un numero intero positivo. **Nota:** il valore "n" deve essere scelto in modo che il periodo di scansione risultante sia un numero intero positivo compreso nell'intervallo [2 - 150 ms].

Esempio di curva di risposta del processo

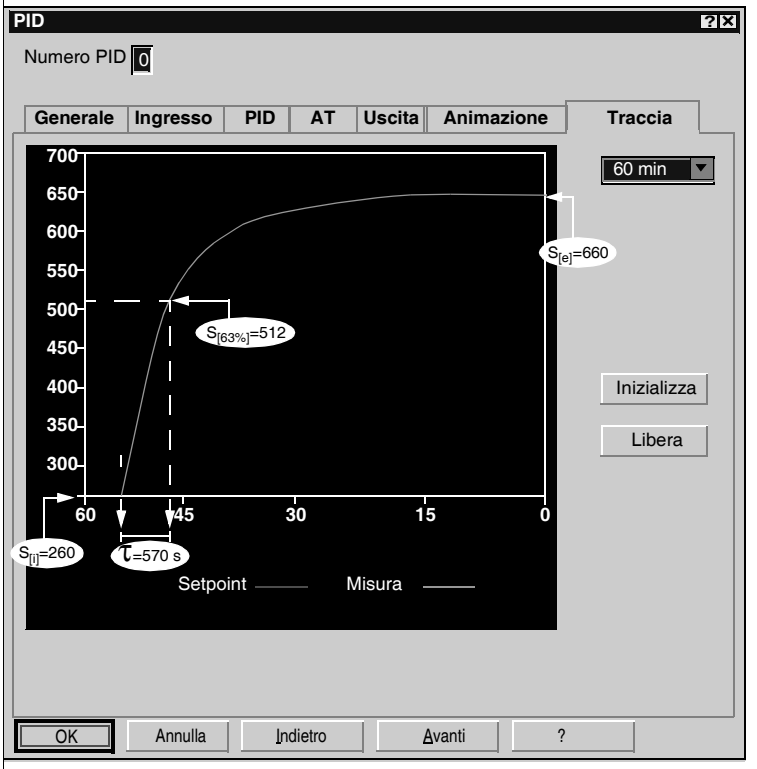
Questo esempio mostra come si misura la costante di tempo (τ) di un processo termico semplice usando il metodo della curva di risposta del processo descritto nella sottosezione precedente.

La configurazione sperimentale per la misura della costante di tempo è la seguente:

- Il processo di controllo consiste in un forno a circolazione forzata di aria equipaggiato di una lampada riscaldante.
- Le misure della temperatura sono raccolte dal controller Twido tramite sonda Pt100 e i dati della temperatura sono registrati in °C.
- Il controller Twido pilota una lampada riscaldante tramite l'uscita discreta PWM del PID.

L'esperimento viene eseguito nel seguente modo:

Passo	Azione
1	Si seleziona la scheda PID **Uscita** nella schermata di configurazione PID.
2	**Si seleziona la Modalità manuale** nella scheda Uscita.
3	Si imposta l'**Uscita** della modalità manuale a 10000.
4	Si avvia l'esecuzione di PID dalla scheda **Traccia.**
5	L'esecuzione di PID viene interrotta quando la temperatura del forno raggiunge uno stato costante.

Passo	Azione
6	Le informazioni seguenti vengono ottenute direttamente dall'analisi grafica della curva di risposta, come mostrato nella figura riportata di seguito:  dove • $S_{[i]}$ = valore iniziale della variabile di processo = 260 • $S_{[e]}$ = valore finale della variabile di processo = 660 • $S_{[63\%]}$ = variabile di processo all'aumento del 63% = $S_{[i]} + (S_{[i]} - S_{[e]}) \times 63\%$ = 260+(660-260)x63% = 512 • τ = costante di tempo = tempo trascorso dall'inizio dell'aumento finché non viene raggiunto $S_{[63\%]}$ = 9 min 30 s = 570 s
7	Il periodo di campionamento (Ts) viene determinato usando la relazione seguente: $Ts = \tau/75$ = 570/75 = 7,6 s (7600 ms)

Passo	Azione
8	Nella finestra di dialogo **Programma > Modifica modalità di scansione**, il **Periodo di scansione** deve essere impostato in modo che il periodo di campionamento (Ts) sia un multiplo esatto del periodo di scansione, come nell'esempio seguente: Periodo di scansione = Ts/76 = 7600/76 = 100 ms (il che soddisfa la condizione: 2 ms ≤ Periodo di scansione ≤ 150 ms)

Metodo della ricerca per tentativi

Il metodo della ricerca per tentativi consiste nel fornire ipotesi successive del periodo di campionamento alla funzione di sintonizzazione automatica finché l'algoritmo di sintonizzazione automatica non converge verso parametri Kp, Ti e Td che vengono ritenuti soddisfacenti dall'utente.

Nota: a differenza del metodo della curva di risposta del processo, il metodo della ricerca per tentativi non è basato su alcuna legge di approssimazione della risposta del processo. Presenta tuttavia il vantaggio di convergere verso un valore del periodo di campionamento che è dello stesso ordine di grandezza del valore attuale.

Per eseguire una stima dei parametri di sintonizzazione automatica con il metodo della ricerca per tentativi, procedere nel seguente modo:

Passo	Azione
1	Selezionare la scheda **AT** nella finestra di configurazione PID.
2	Impostare la **Limitazione di uscita** di AT a **10000.**
3	Selezionare la scheda **PID** nella finestra di configurazione del PID.
4	Immettere la prima o la ennesima ipotesi nel campo **Periodo di campionamento**. **Nota:** se non si ha alcuna indicazione circa l'intervallo possibile per il periodo di campionamento, impostare questo valore al minimo possibile: 1 (1 unità di 10 ms).
5	Selezionare **Controller > Trasferimento PC => Controller...** dalla barra dei menu per scaricare il programma dell'applicazione nel controller Twido.
6	Avviare **Sintonizzazione automatica.**
7	Selezionare la scheda **Animazione** nella schermata di configurazione PID.
8	Attendere il termine del processo di sintonizzazione automatica.

Passo	Azione
9	Possono verificarsi due casi: ● **La sintonizzazione viene eseguita correttamente:** si può continuare con il passo 9. ● **La sintonizzazione automatica non riesce:** questo significa che l'ipotesi corrente per il periodo di campionamento (Ts) non è corretta. Provare con una nuova ipotesi di Ts e ripetere i passi da 3 a 8 fino all'eventuale conversione del processo di sintonizzazione automatica. Attenersi alle istruzioni seguenti per fornire una nuova ipotesi di Ts: ● AT termina con il messaggio di errore **"La costante di tempo calcolata è negativa":** Questo significa che il periodo di campionamento **Ts è troppo lungo.** Occorre diminuire il valore di Ts per fornire una nuova ipotesi. ● AT termina con il messaggio di errore **"Errore di campionamento":** Questo significa che il periodo di campionamento **Ts è troppo breve.** Occorre aumentare il valore di Ts per fornire una nuova ipotesi.
10	I parametri di controllo PID (Kp, Ti e Td) possono ora essere visualizzati nella scheda Animazione, quindi regolati nella scheda PID della schermata di configurazione PID, come necessario. **Nota:** se la regolazione PID fornita da questo insieme di parametri di controllo non dà risultati completamente soddisfacenti, si può perfezionare la stima del periodo di campionamento tramite la ricerca per tentativi fino a ottenere l'insieme corretto di parametri di controllo Kp, Ti e Td.

Regolazione di parametri PID

Per perfezionare la regolazione di processo fornita dai parametri PID (Kp, Ti, Td) ottenuti con la sintonizzazione automatica, è anche possibile regolare i valori dei parametri manualmente, direttamente nella scheda PID della schermata di configurazione PID o tramite le parole di memoria corrispondenti (%MW).

Limitazioni relative all'uso della sintonizzazione automatica e del controllo PID

La **sintonizzazione automatica** è particolarmente adatta per processi in cui la costante di tempo (τ) e il ritardo (θ) soddisfano il requisito seguente: ($\tau + \theta$) < 2700 s (ossia: 45 min)
Il **controllo PID** è adatto per la regolazione di processi che soddisfano la condizione seguente: 2 < (τ/θ) < 20, dove (τ) è la costante di tempo del processo e (θ) il ritardo.

Nota: a seconda del rapporto (τ/θ):
- (τ/θ) < 2: La regolazione PID ha raggiunto i suoi limiti; in questo caso sono necessarie tecniche di regolazione più avanzate.
- (τ/θ) > 20: In questo caso può essere usato un semplice controller on/off (o a due posizioni) al posto del controller PID.

Risoluzione dei problemi della funzione di sintonizzazione automatica

La tabella seguente registra i messaggi di errore della funzione di sintonizzazione automatica e descrive le cause e i rimedi possibili:

Messaggio di errore	Causa possibile	Spiegazione/Soluzione possibile
Errore di sintonizzazione automatica: raggiunto limite di variabile di processo (PV)	La variabile di processo ha raggiunto il valore massimo consentito.	Questo è un messaggio di sicurezza del sistema. Dato che la funzione AT è un processo a loop aperto, il limite della variabile di processo funge da limite massimo.
Errore di sintonizzazione automatica: campionamento eccessivo o setpoint di uscita impostato troppo basso	Due cause possibili: ● Periodo di campionamento troppo breve. ● Uscita AT impostata troppo bassa.	Aumentare il periodo di campionamento o valore di setpoint dell'uscita AT.
Errore di sintonizzazione automatica: costante di tempo negativa	Il periodo di campionamento può essere troppo lungo.	Per ulteriori informazioni, vedere *Sintonizzazione del PID con sintonizzazione automatica (Auto-Tuning, AT), p. 553*.
Errore di sintonizzazione automatica: errore nel calcolo di Kp	L'algoritmo AT non è riuscito (nessuna convergenza).	Controllare i parametri PID e AT e modificarli in modo da migliorare la convergenza. Verificare inoltre che non esistano disturbi che possano influenzare la variabile di processo.
Errore di sintonizzazione automatica: costante di tempo superiore a rapporto di ritardo > 20	$\tau/\theta > 20$	La regolazione PID non è più garantita. Per ulteriori informazioni, vedere *Sintonizzazione del PID con sintonizzazione automatica (Auto-Tuning, AT), p. 553*.
Errore di sintonizzazione automatica: costante di tempo superiore a rapporto di ritardo < 2	$\tau/\theta < 2$	La regolazione PID non è più garantita. Per ulteriori informazioni, vedere *Sintonizzazione del PID con sintonizzazione automatica (Auto-Tuning, AT), p. 553*.
Errore di sintonizzazione automatica: limite per Kp superato	Il valore calcolato del guadagno statico (Kp) è maggiore di 10000.	La sensibilità di misura di alcune variabili di applicazione può essere troppo bassa. Il campo di misura dell'applicazione deve essere scalato nuovamente entro l'intervallo [0-10000].
Errore di sintonizzazione automatica: limite per Ti superato	Il valore calcolato della costante di tempo integrale (Ti) è maggiore di 20000.	È stato raggiunto il limite di calcolo.
Errore di sintonizzazione automatica: limite per Td superato	Il valore calcolato della costante di tempo derivativa (Td) è maggiore di 10000.	È stato raggiunto il limite di calcolo.

Metodo di regolazione dei parametri PID

Introduzione

I metodi che consentono di regolare i parametri PID sono numerosi, in questa sede si suggeriscono i metodi Ziegler e Nichols, che presentano due varianti:

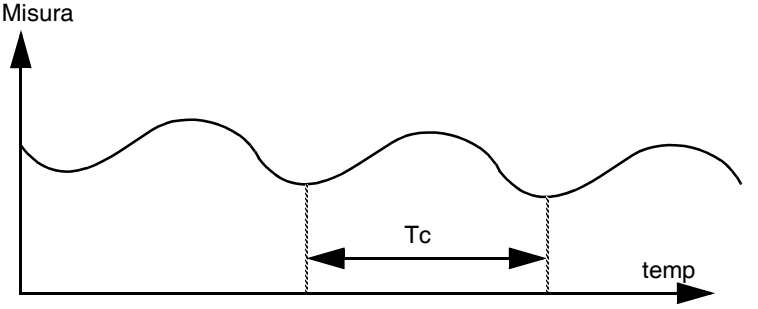
- regolazione a loop chiuso,
- regolazione a loop aperto.

Prima di implementare questi metodi, è necessario stabilire la direzione dell'azione del PID:

- se un aumento dell'uscita OUT provoca un incremento della misura PV, si ha un PID invertito (KP > 0),
- se, al contrario, si verifica una riduzione di PV, si ha un PID diretto (KP < 0).

Regolazione a loop chiuso

Il principio su cui si basa utilizza un comando proporzionale (Ti = 0, Td = 0) per avviare il processo, aumentando la produzione fino a quando inizia a oscillare nuovamente, dopo l'applicazione di un livello al setpoint del correttore PID. È sufficiente aumentare del livello critico di produzione (Kpc) che ha provocato l'oscillazione non mitigata e il periodo di oscillazione (Tc) per ridurre i valori e ottenere un'impostazione ottimale del regolatore.



A seconda del tipo di regolatore (PID o PI), l'impostazione dei coefficienti avviene tramite i seguenti valori:

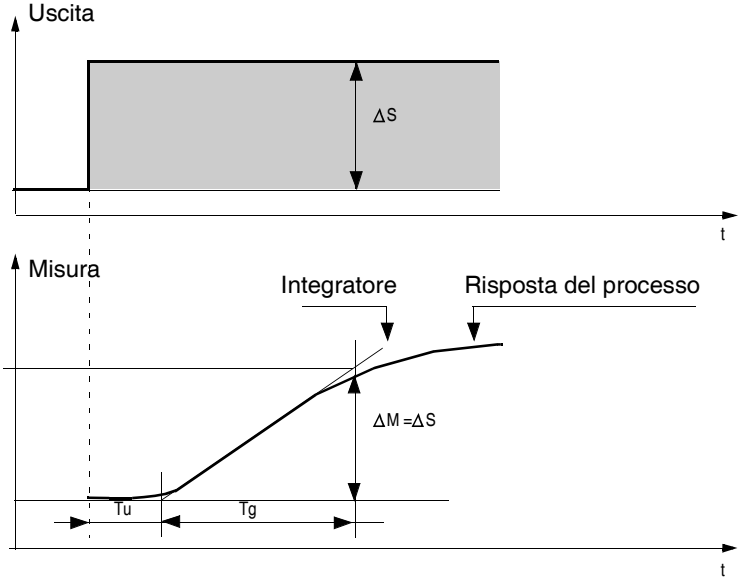
-	Kp	Ti	Td
PID	Kpc/1,7	Tc/2	Tc/8
PI	Kpc/2,22	0,83 x Tc	-

dove Kp = produzione proporzionale, Ti = tempo di integrazione e TD = tempo di diversione.

Nota: questo metodo di regolazione fornisce un controllo molto dinamico che può provocare picchi indesiderati in corrispondenza del cambiamento degli impulsi di setpoint. In tal caso, è opportuno ridurre il valore di produzione fino a ottenere il comportamento desiderato.

Regolazione a loop aperto

Poiché il regolatore si trova in modalità manuale, si applica un livello all'uscita si avvia la risposta della procedura come un integratore con puro ritardo.



Il punto di intersezione sul lato destro rappresenta l'integratore con gli assi del tempo determina il tempo Tu. In seguito, il tempo Tg è definito come il tempo necessario perché la variabile controllata (misurazione) abbia una variazione di dimensioni uguali (% della scala) dell'uscita regolatore.
A seconda del tipo di regolatore (PID o PI), l'impostazione dei coefficienti avviene tramite i seguenti valori:

-	**Kp**	**Ti**	**Td**
PID	-1,2 Tg/Tu	2 x Tu	0,5 x Tu
PI	-0,9 Tg/Tu	3,3 x Tu	-

dove Kp = produzione proporzionale, Ti = tempo di integrazione e TD = tempo di diversione.

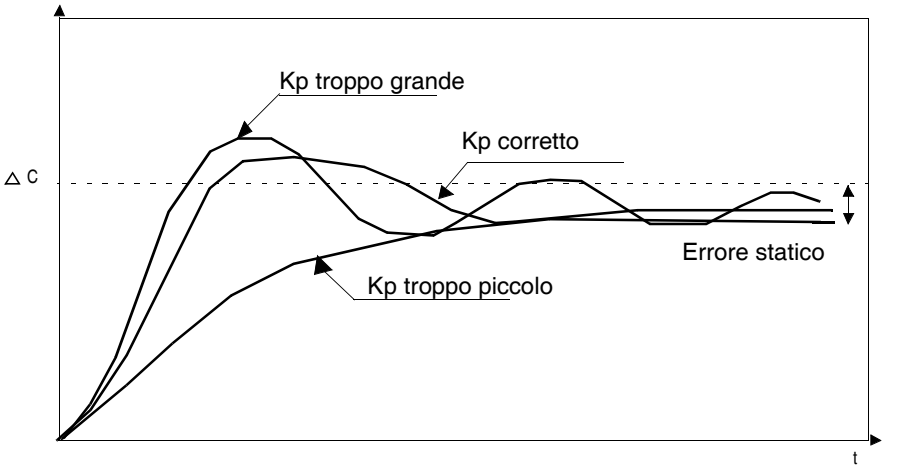
Nota: prestare attenzione alle unità. Se la regolazione è eseguita in PL7, moltiplicare per 100 il valore ottenuto per KP.

Anche questo metodo di regolazione fornisce un controllo molto dinamico che può provocare picchi indesiderati in corrispondenza del cambiamento degli impulsi di setpoint. In tal caso, è opportuno ridurre il valore di produzione fino a ottenere il comportamento desiderato. Questo metodo è interessante, perché non richiede di formulare ipotesi sulla natura e sull'ordine della procedura. È possibile infatti applicarlo sia a procedure stabili sia a procedure di integrazione. È particolarmente adatto, inoltre, nel caso di procedure lente (ad esempio nel settore vetrario,…) perché all'utente è sufficiente solo l'inizio della risposta per regolare i coefficienti Kp, Ti e Td.

Ruolo e influenza dei parametri di un PID

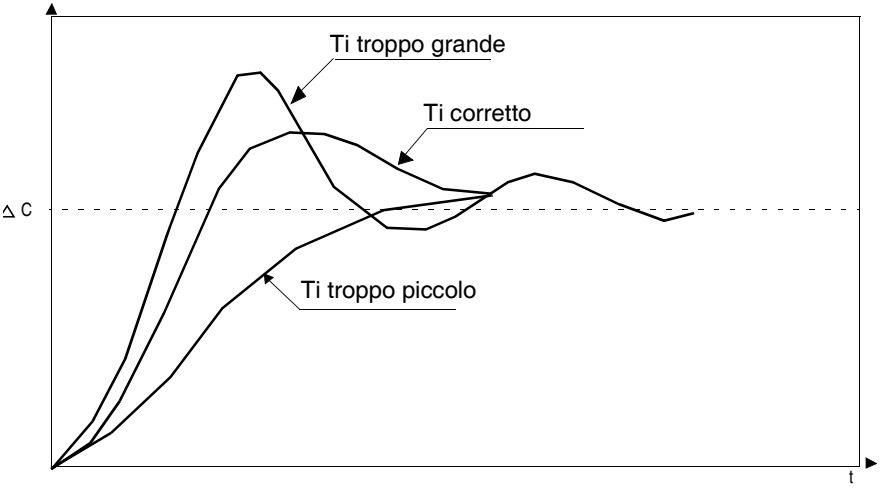
Influenza dell'azione proporzionale

L'azione proporzionale permette di influire sia sulla velocità sia sulla risposta del processo. Più il guadagno è elevato, più la risposta s'accelera, più l'errore statico diminuisce (proporzionalmente), e più la stabilità si degrada. Occorre trovare il miglior compromesso tra velocità e stabilità L'influenza dell'azione integrale sulla risposta del processo a un'unità di avanzamento è la seguente:



Influenza dell'azione integrale

L'azione integrale permette di annullare l'errore statico tra la misura e il setpoint). Più l'azione integrale è elevata (Ti piccolo), più la risposta s'accelera e più la stabilità si degrada. Occorre altresì trovare un buon compromesso tra velocità e stabilità. L'influenza dell'azione integrale sul risposta del processo all'avanzamento di un'unità è la seguente:

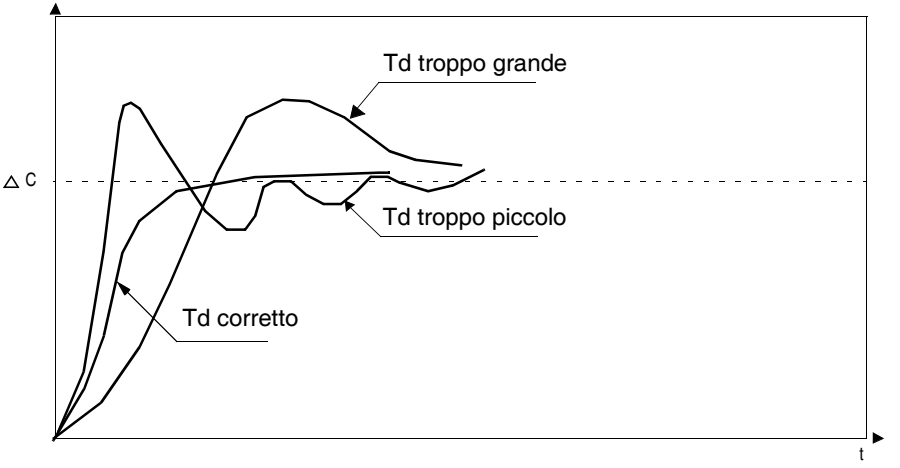


Nota: Ti piccolo signifca un'azione integrale elevata.

dove Kp = guadagno proporzionale, Ti = tempo d'integrazione e Td = tempo di derivazione.

Influenza dell'azione derivata

L'azione derivata è anticipatrice. In effetti, essa aggiunge un termine che tiene conto della velocità di variazione della deviazione, e questo permette d'anticipare in accelerazione la risposta al processo, quando la deviazione cresce, e in decelerazione quando la deviazione diminuisce. Più l'azione derivata è alta (Td grande) più la risposta si accelera. Anche in questo caso, occorre trovare il miglior compromesso tra velocità e stabilità L'influenza dell'azione derivata sulla risposta del processo a un'unità di avanzamento è la seguente:



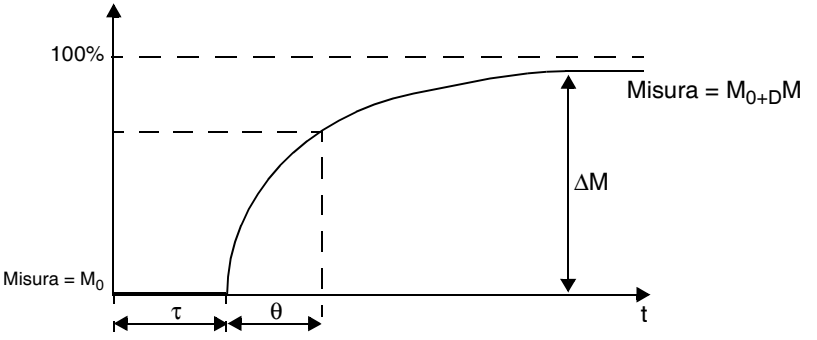
Limiti della regolazione PID

Se si assimila il processo di primo ordine ad un ritardo puro di funzione di trasferimento: $(H(p)) = K\frac{(e^{(-\tau)p})}{(1+\theta p)}$

con:

τ =ritardo del modello,

θ = costante del tempo del modello,



I risultati della regolazione dipendono dal rapporto $\frac{\tau}{\theta}$

La regolazione PID è particolarmente adatta nei seguenti settori:2- $\frac{\tau}{\theta}$ -20

Per $\frac{\tau}{\theta}$ <2, ossia dei loop veloci (θ piccoli) o dei processi con tempi di risposta troppo grandi (t grande) la regolazione PID non è più interessante, occorre utilizzare degli algoritmi più complessi.

Per $\frac{\tau}{\theta}$ >20, una regolazione a soglia, più isteresi, dovrebbe essere sufficiente.

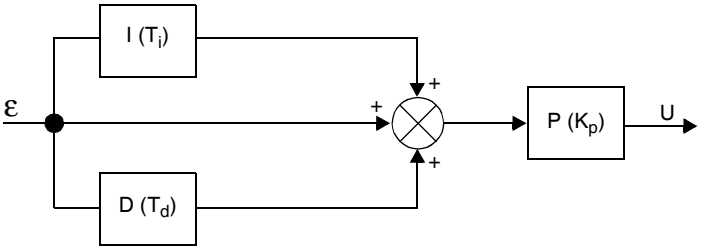
Appendice 1: Fondamenti della teoria PID

Introduzione

La funzione di controllo PID presente su tutti i controller Twido fornisce un controllo efficiente di processi industriali semplici che consiste in uno stimolo di sistema (definito setpoint in questo documento) e in una proprietà misurabile del sistema (definita misura o variabile di processo).

Modello di controller PID

Il controller PID Twido realizza una correzione PID mista (seriale - parallela) a partire da una misura e da un setpoint analogici nel formato [0 - 10000] e fornisce un comando analogico al processo controllato nello stesso formato (vedere il Diagramma del modello PID riportato di seguito).
La forma mista del modello di controller PID è descritta nel diagramma seguente:



dove
dove:

- I = l'azione **integrale** (che è indipendente e parallela all'azione derivativa),
- D = l'azione **derivativa** (che è indipendente e parallela all'azione integrale),
- P = l'azione **proporzionale** (che funziona in modo seriale sull'uscita combinata delle azioni integrale e derivativa),
- U = uscita del controller PID (in seguito ingresso del processo controllato).

Legge di controllo PID

Il controller PID è composto dalla combinazione mista (seriale - parallela) di guadagno del controller (Kp) e dalle costanti di tempo integrale (Ti) e derivativa (Td). Pertanto la legge di controllo PID usata dal controller Twido ha la forma seguente *(Eq.1):*

$$u(i) = K_P \cdot \left\{ \varepsilon(i) + \frac{T_s}{T_i} \sum_{j=1}^{i} \varepsilon(j) + \frac{T_d}{T_s}[\varepsilon(i) - \varepsilon(i-1)] \right\}$$

dove

- Kp = il guadagno proporzionale del controller,
- Ti = la costante di tempo integrale,
- Td = la costante di tempo derivativa,
- Ts = il periodo di campionamento,
- $\varepsilon(i)$ = la deviazione ($\varepsilon(i)$ = setpoint - variabile di processo).

Nota: vengono usati due algoritmi di calcolo diversi, a seconda del valore della costante di tempo integrale (Ti):

- Ti ≠ 0: In questo caso viene usato un algoritmo incrementale.
- Ti = 0: Questo è un processo non integrante. Qui viene usato un algoritmo posizionale, insieme a un offset +5000 applicato alla variabile di uscita PID.

Per una descrizione dettagliata di Kp, Ti e Td, vedere *Scheda PID del PID, p. 533*. Come si può dedurre da *(equ.1)* e *(equ.1')*, il parametro chiave per la regolazione PID è il **periodo di campionamento (Ts)**. Il periodo di campionamento dipende essenzialmente dalla **costante di tempo (τ)**, un parametro intrinseco al processo che si desidera controllare con la funzione PID (vedere l'*Appendice 2: Modello di prim'ordine con ritardo, p. 570*).

Appendice 2: Modello di prim'ordine con ritardo

Introduzione

Questa sezione presenta il modello di prim'ordine con ritardo usato per descrivere vari processi industriali semplici ma importanti, tra cui i processi termici.

Modello di prim'ordine con ritardo

È ampiamente accettato il fatto che i processi termici semplici (a uno stimolo) possano essere approssimati adeguatamente con un modello di prim'ordine con ritardo.
La funzione di trasferimento di questo processo di prim'ordine a ciclo aperto ha la forma seguente nel dominio di Laplace *(equ.2):*

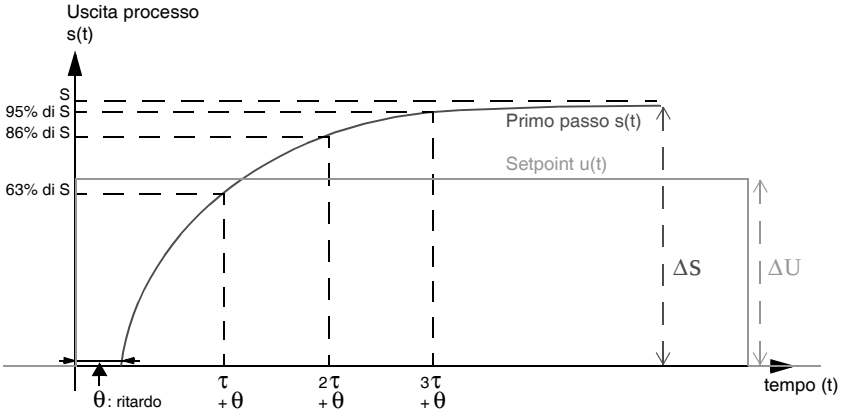
$$\frac{S}{U} = \frac{k}{1 + \tau p} \cdot e^{-\theta p}$$

dove

- k = guadagno statico,
- τ = costante di tempo,
- θ = ritardo,
- U = ingresso di processo (uscita del controller PID),
- S = uscita di processo.

Costante di tempo di processo τ

Il parametro chiave della legge di risposta di processo *(equ.2)* è la **costante di tempo** τ**.** Questo parametro è intrinseco al processo da controllare.
La costante di tempo (τ) di un sistema di prim'ordine è definita come il tempo (in sec) in cui la variabile di uscita di sistema raggiunge il 63% dell'uscita finale dal momento in cui il sistema ha iniziato a reagire allo stimolo a gradino u(t).
La figura seguente mostra una tipica risposta di processo di prim'ordine a uno stimolo a gradino:



dove

- k = guadagno statico calcolato come il rapporto $\Delta S/\Delta U$,
- τ = tempo all'aumento 63% = costante di tempo,
- 2τ = tempo all'aumento 86%,
- 3τ = tempo all'aumento 95%.

Nota: quando la sintonizzazione automatica è implementata, il periodo di campionamento (Ts) deve essere scelto nel seguente intervallo: [$\tau/125 < Ts < \tau/25$]. Idealmente si dovrebbe usare [$Ts= \tau/75$] (vedere *Sintonizzazione del PID con sintonizzazione automatica (Auto-Tuning, AT), p. 553*).

17.5 Istruzioni su virgola mobile

In breve

Oggetto di questa sezione

Questo capitolo descrive le istruzioni avanzate sulla virgola mobile (Vedi *Oggetti a virgola mobile e parola doppia, p. 32*) per il linguaggio TwidoSoft.
Le istruzioni di confronto e di assegnazione sono descritte alla sezione *Elaborazione numerica, p. 415*

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Istruzioni aritmetiche in virgola mobile	573
Istruzioni trigonometriche	576
Istruzioni di conversione	578
Istruzioni di conversione interi <-> Virgola mobile	580

Istruzioni aritmetiche in virgola mobile

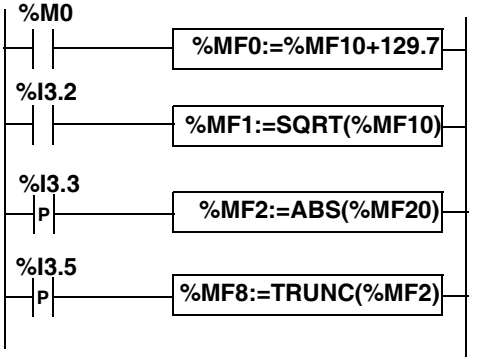
Punti generali

Le seguenti istruzioni consentono di eseguire un'operazione aritmetica tra due operandi o su un operando.

+	addizione di due operandi	**SQRT**	radice quadrata di un operando
-	sottrazione di due operandi	**ABS**	valore assoluto di un operando
*****	moltiplicazione di due operandi	**TRUNC**	parte intera di un valore in virgola mobile
/	divisione di due operandi	**EXP**	esponenziale naturale
LOG	logaritmo in base 10	**EXPT**	Potenza di un intero con numero reale
LN	logaritmo neperiano		

Struttura

Ladder



Linguaggio Lista d'istruzioni

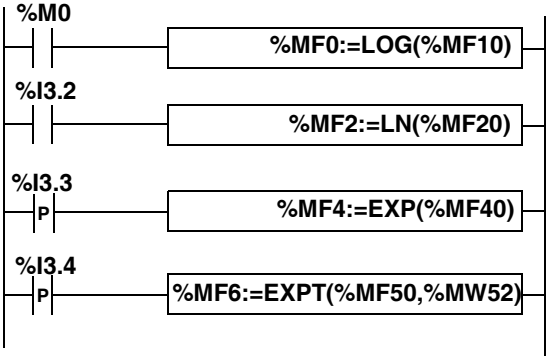
```
LD %M0
[%MF0:=%MF10+129.7]

LD %I3.2
[%MF1:=SQRT(%MF10)]

LDR %I3.3
[%MF2:=ABS(%MF20)]

LDR %I3.5
[%MF8:=TRUNC(%MF2)]
```

Linguaggio Ladder



Linguaggio List

```
LD %M0
[%MF0:=LOG(%MF10]

LD %I3.2
[%MF2:=LN(%MF20)]

LDR %I3.3
[%MF4:=EXP(%MF40)]

LDR %I3.4
[%MF6:=EXPT(%MF50,%MW52)]
```

Sintassi

Operatori e sintassi delle istruzioni aritmetiche in virgola mobile

Operatori	Sintassi
+, - *, /	Op1:=Op2 Operatore Op3
SQRT, ABS, TRUNC, LOG, EXP, LN	Op1:=Operatore(Op2)
EXPT	Op1:=Operatore (Op2,Op3)

Nota: quando si esegue un'addizione o una sottrazione tra due numeri a virgola mobile, i due operandi devono rispettare la condizione $\mathrm{Op1} > \mathrm{Op2} \times 2^{-24}$ con Op1>Op2. Se questa condizione non viene rispettata, il risultato sarà uguale all'operando 1 (Op1). Questo comportamento non ha conseguenze di rilievo se si tratta di un'operazione isolata, poiché l'errore risultante è minimo (2^{-24}), ma ha esiti imprevisti nel caso di un'operazione ripetitiva.
Es: si consideri l'istruzione **%MF2:= %MF2 + %MF0** ripetuta all'infinito. Se le condizioni iniziali sono %MF0 = 1.0 e %MF2= 0, si osserva che il valore %MF2 si blocca a 16777216. Si consiglia dunque di tenere conto delle operazioni ripetitive durante la programmazione. Se si desidera tuttavia programmare questo tipo di calcolo, è compito del programma di applicazione dell'utente gestire gli errori di troncatura.

Operandi delle istruzioni aritmetiche in virgola mobile:

Operatori	Operando 1 (Op1)	Operando 2 (Op2)	Operando 3 (Op3)
+, - *, /	%MFi	%MFi, %KFi, valore immediato	%MFi, %KFi, valore immediato
SQRT, ABS, LOG, EXP, LN	%MFi	%MFi, %KFi	[-]
TRUNC	%MFi	%MFi, %KFi	[-]
EXPT	%MFi	%MFi, %KFi	%MWi, %KWi, valore immediato

Regole d'utilizzo

- non è possibile eseguire operazioni direttamente tra numeri a virgola mobile e numeri interi. Le operazioni di conversione (Vedi *Istruzioni di conversione interi <-> Virgola mobile, p. 580*) assicurano la conversione in uno dei due formati.
- il bit di sistema %S18 viene gestito in modo identico alle operazioni su numeri interi (Vedi *Istruzioni aritmetiche su interi, p. 424*), la parola %SW17 (Vedi *Parole di sistema (%SW), p. 614*) indica la causa di errore.
- quando l'operando della funzione è un valore non valido (esempio: logaritmo di un numero negativo), produce un risultato indeterminato o infinito e fa passare il bit %S18 a 1, la parola %SW17 indica la causa dell'errore.

Istruzioni trigonometriche

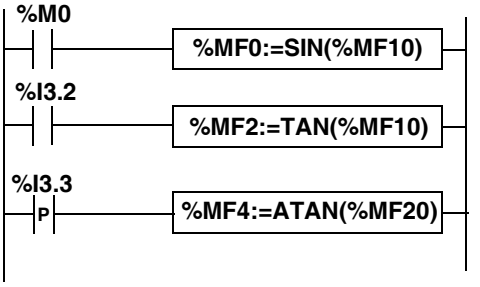
Generale

Queste istruzioni permettono di realizzare operazioni trigonometriche.

SIN	seno di un angolo espresso in radianti,	**ASIN**	arco seno (risultante tra $-\frac{\pi}{2}$ e $\frac{\pi}{2}$)
COS	coseno di un angolo espresso in radianti,	**ACOS**	arco coseno (risultante tra 0 e π)
TAN	tangente di un angolo espressa in radianti,	**ATAN**	arco tangente (risultante tra $-\frac{\pi}{2}$ e $\frac{\pi}{2}$)

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %M0
[%MF0:=SIN(%MF10)]

LD %I3.2
[%MF2:=TAN(%MF10)]

LDR %I3.3
[%MF4:=ATAN(%MF20)]
```

Linguaggio letterale strutturato

```
IF %M0 THEN
 %MF0:=SIN(%MF10);
END_IF;
IF %I3.2 THEN
 %MF2:=TAN(%MF10);
END_IF;
IF %I3.3 THEN
 %MF4:=ATAN(%MF20);
END_IF;
```

Sintassi

Operatori, operandi e sintassi delle istruzioni di operazioni trigonometriche

Operatori	Sintassi	Operando 1 (Op1)	Operando 2 (Op2)
SIN, COS, TAN, ASIN, ACOS, ATAN	Op1:=Operatore(Op2)	%MFi	%MFi, %KFi

Regole d'uso

- quando l'operando della funzione è un valore non valido (esempio: arco coseno di un numero maggiore di 1), produce un risultato indeterminato o infinito e fa passare il bit %S18 a 1, la parola %SW17 (Vedi *Parole di sistema (%SW), p. 614*) indica la causa dell'errore.
- le funzioni SIN/COS/TAN ammettono nei parametri un angolo tra -4096π e 4096π ma la precisione decresce progressivamente per gli angoli fuori dall'intervallo -2π e $+2\pi$ a causa dell'imprecisione apportata dal modulo 2π effettuato sul parametro prima di tutte le operazioni.

Istruzioni di conversione

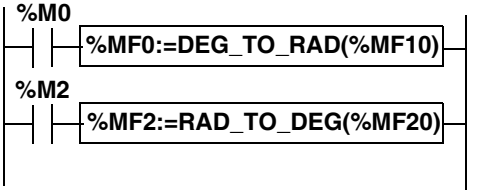
Generale

Queste istruzioni permettono di realizzare operazioni di conversione.

DEG_TO_RAD	conversione di gradi in radianti, il risultato è il valore dell'angolo compreso tra 0 e 2π
RAD_TO_DEG	conversione di un angolo espresso in radianti, il risultato è il valore dell'angolo compreso tra 0 e 360 gradi

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %M0
[%MF0:=DEG_TO_RAD(%MF10)]

LD %M2
[%MF2:=RAD_TO_DEG(%MF20)]
```

Linguaggio letterale strutturato

```
IF %M0 THEN
 %MF0:=DEG_TO_RAD(%MF10);
END_IF;
IF %M2 THEN
 %MF2:=RAD_TO_DEG(%MF20);
END_IF;
```

Sintassi

Operatori, operandi e sintassi delle istruzioni di conversione

Operatori	Sintassi	Operando 1 (Op1)	Operando 2 (Op2)
DEG_TO_RAD **RAD_TO_DEG**	Op1:=Operatore(Op2)	%MFi	%MFi, %KFi

Regole d'uso

L'angolo da convertire deve essere compreso tra -737280.0 e +737280.0 (per le conversioni DEG_TO_RAD) o tra -4096π e 4096π (per le conversioni RAD_TO_DEG).
Per i valori non compresi tra questi limiti il risultato visualizzato sarà + 1.#NAN, essendo i bit %S18 e %SW17:X0 impostati a 1.

Istruzioni di conversione interi <-> Virgola mobile

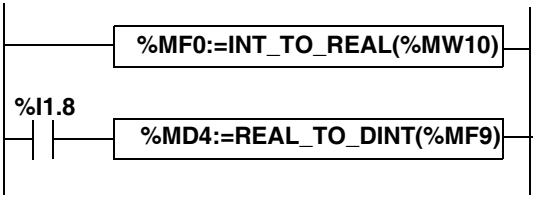
Generale

Sono proposte quattro istruzioni di conversione.
Elenco delle istruzioni di conversione interi<-> virgola mobile:

INT_TO_REAL	conversione parola di interi --> virgola mobile
DINT_TO_REAL	conversione di parola doppia (intero) --> virgola mobile
REAL_TO_INT	conversione di virgola mobile --> parola di interi. Il risultato è il valore algebrico più vicino.
REAL_TO_DINT	conversione di virgola mobile --> parola doppia di interi. Il risultato è il valore algebrico più vicino.

Struttura

Linguaggio Ladder



Linguaggio List

```
LD TRUE
[%MF0:=INT_TO_REAL(%MW10)]

LD I1.8
[%MD4:=REAL_TO_DINT(%MF9)]
```

Linguaggio letterale strutturato

```
%MF0:=INT_TO_REAL(%MW10);
IF %I1.8 THEN
 %MD4:=REAL_TO_DINT(%MF9);
END_IF;
```

Sintassi

Operatori e sintassi (conversione parola di interi --> virgola mobile):

Operatori	Sintassi
INT_TO_REAL	Op1=INT_TO_REAL(Op2)

Operandi (conversione parola di interi --> virgola mobile):

Operando 1 (Op1)	Operando 2 (Op2)
%MFi	%MWi,%KWi

Esempio: conversione parola di interi --> virgola mobile: 147 --> 1.47e+02

Operatori e sintassi (conversione doppia parola di interi --> virgola mobile):

Operatori	Sintassi
DINT_TO_REAL	Op1=DINT_TO_REAL(Op2)

Operandi (conversione doppia parola di interi --> virgola mobile):

Operando 1 (Op1)	Operando 2 (Op2)
%MFi	%MDi,%KDi

Esempio:conversione parola doppia di interi --> virgola mobile: 68905000 --> 6.8905e+07

Operatori e sintassi (conversione virgola mobile --> parola o parola doppia di interi):

Operatori	Sintassi
REAL_TO_INT	Op1=Operatore(Op2)
REAL_TO_DINT	

Operandi (conversione virgola mobile --> parola o parola doppia di interi):

Tipo	Operando 1 (Op1)	Operando 2 (Op2)
Parole	%MWi	%MFi, %KFi
Parole doppie	%MDi	%MFi, %KFi

Esempio:
conversione virgola mobile --> parola di interi: 5978.6 --> 5979
conversione virgola mobile --> parola doppia di interi: -1235978.6 --> -1235979

Nota: Se durante una conversione da numero reale a intero o a parola doppia di interi il valore in virgola mobile è esterno ai limiti della parola o parola doppia, il bit %S18 è impostato a 1.

Precisione di arrotondamento

La norma IEEE 754 definisce 4 modalità di arrotondamento per le operazioni a virgola mobile.
La modalità utilizzata dalle istruzioni sopra riportate è "arrotondamento al valore più vicino": "se i più vicini valori rappresentabili sono ad eguale distanza dal risultato teorico, il valore fornito sarà quello il cui bit meno rappresentativo è pari a 0".
In determinati casi, il risultato dell'arrotondamento può dunque assumere un valore per difetto o un valore per eccesso.

Ad esempio:
Arrotondamento del valore 10,5 -> 10
Arrotondamento del valore 11,5 -> 12

17.6 Istruzioni sulle tabelle di oggetti

In breve

Oggetto di questa sezione

Questa sezione descrive le istruzioni specifiche relative alle tabelle:
- di parole doppie,
- di virgole mobili

Le istruzioni di assegnazione sulle tabelle sono descritte al capitolo relativo alle istruzioni di base (Vedi *Assegnazione di tabelle di parole, parole doppie o virgole mobili, p. 420*).

Contenuto di questa sezione

Questo sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Funzione di somma sulle tabelle	584
Funzione di confronto delle tabelle	586
Funzioni di ricerca sulle tabelle	588
Funzioni di ricerca del valore massimo e minimo sulle tabelle	590
Numero di occorrenze di un valore in una tabella	591
Funzione di scorrimento circolare su una tabella	592
Funzione di ordinamento sulle tabelle	594
Funzione di interpolazione su una tabella a virgola mobile	596
Funzione di calcolo della media dei valori di una tabella a virgola mobile	601

Funzione di somma sulle tabelle

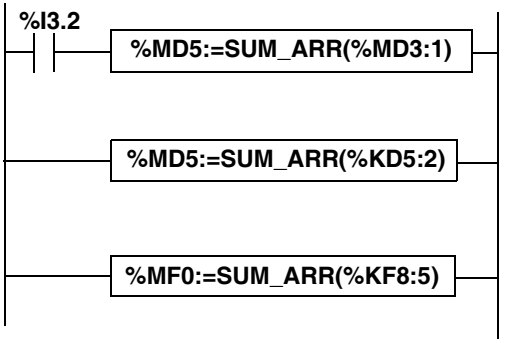
Punti generali

La funzione SUM_ARR esegue la somma di tutti gli elementi di una tabella di oggetti:

- se la tabella è costituita da parole doppie, il risultato viene dato sotto forma di una parola doppia,
- se la tabella è costituita da parole a virgola mobile, il risultato viene dato sotto forma di una parola a virgola mobile.

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %I3.2
[%MD5:=SUM_ARR(%MD3:1)]
%MD5:=SUM_ARR(%KD5:2)
%MF0:=SUM_ARR(%KF8:5)
```

Sintassi

Sintassi dell'istruzione di somma sulle tabelle:

```
Ris:=SUM_ARR(Tab)
```

Parametri dell'istruzione di somma sulle tabelle

Tipo	Risultato (Ris)	Tabella (Tab)
Tabelle di parole doppie	%MDi	%MDi:L,%KDi:L
Tabelle in virgola mobile	%MFi	%MFi:L,%KFi:L

Nota: il bit %S18 viene impostato a 1 quando il risultato non è nei limiti del formato parola doppia secondo l'operando di tabella.

Esempio

```
%MD5:=SUM(%MD30:4)
```

con %MD30=10, %MD31=20, %MD32=30, %MD33=40
%MD5=10+20+30+40=100

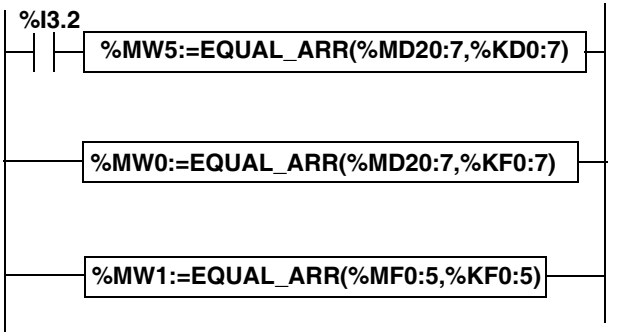
Funzione di confronto delle tabelle

Punti generali

La funzione EQUAL _ARR esegue il confronto di due tabelle elemento per elemento.
Se appare una differenza, la posizione dei primi elementi differenti viene restituita in forma di parola, altrimenti il valore restituito sarà uguale a -1.
Il confronto è eseguito sulla totalità della tabella.

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %I3.2
[%MW5:=EQUAL_ARR(%MD20:7,KD0:7)]
```

Linguaggio letterale strutturato

```
%MW0:=EQUAL_ARR(%MD20:7,%KF0:7)

%MW1:=EQUAL_ARR(%MF0:5,%KF0:5)
```

Sintassi

Sintassi dell'istruzione di confronto sulle tabelle:

```
Ris:=EQUAL_ARR(Tab1,Tab2)
```

Parametri delle istruzioni di confronto sulle tabelle

Tipo	Risultato (Ris)	Tabelle (Tab1 e Tab2)
Tabelle di parole doppie	%MWi	%MDi:L,%KDi:L
Tabelle in virgola mobile	%MWi	%MFi:L,%KFi:L

Nota:
- le tabelle devono essere obbligatoriamente della stessa lunghezza e dello stesso tipo

Esempio

```
%MW5:=EQUAL_ARR(%MD30:4,%KD0:4)
```

Confronto di due tabelle:

Posizione	Tabelle di parole	Tabelle di costanti	Differenza
0	%MD30=10	%KD0=10	=
1	%MD31=20	%KD1=20	=
2	%MD32=30	%KD2=60	Differente
3	%MD33=40	%KD3=40	=

La parola %MW5 ha valore 2 (prima posizione differente)

Funzioni di ricerca sulle tabelle

Punti generali

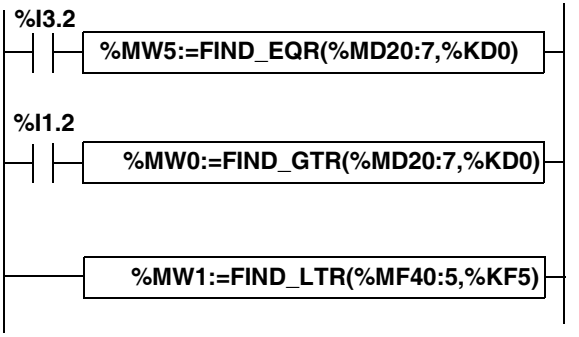
Sono proposte 3 funzioni di ricerca:

- **FIND_EQR** : ricerca della posizione in una tabella di parole doppie o virgole mobili del primo elemento uguale a un valore dato
- **FIND_GTR** : ricerca della posizione in una tabella di parole doppie o virgole mobili del primo elemento superiore a un valore dato
- **FIND_LTR** : ricerca della posizione in una tabella di parole doppie o virgole mobili del primo elemento inferiore a un valore dato

Il risultato di queste istruzioni è uguale alla posizione del primo elemento trovato o a -1 se la ricerca è infruttuosa.

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %I3.2
[%MW5:=FIND_EQR(%MD20:7,KD0)]
LD %I1.2
[%MW0:=FIND_GTR(%MD20:7,%KD0)]
%MW1:=FIND_LTR(%MF40:5,%KF5)
```

Sintassi

Sintassi delle istruzioni di ricerca sulle tabelle:

Funzione	Sintassi
FIND_EQR	Ris:=Funzione(Tab,Val)
FIND_GTR	
FIND_LTR	

Parametri delle istruzioni di ricerca su tabelle di virgole mobili e parole doppie:

Tipo	Risultato (Ris)	Tabella (Tab)	Valore (val)
Tabelle in virgola mobile	%MWi	%MFi:L,%KFi:L	%MFi,%KFi
Tabelle di parole doppie	%MWi	%MDi:L,%KDi:L	%MDi,%KDi

Esempio

```
%MW5:=FIND_EQR(%MD30:4,%KD0)
```

Ricerca della posizione della prima parola doppia =%KD0=30 nella tabella:

Posizione	Tabelle di parole	Risultato
0	%MD30=10	-
1	%MD31=20	-
2	%MD32=30	%MW5=2 (valore della posizione)
3	%MD33=40	-

Funzioni di ricerca del valore massimo e minimo sulle tabelle

Punti generali

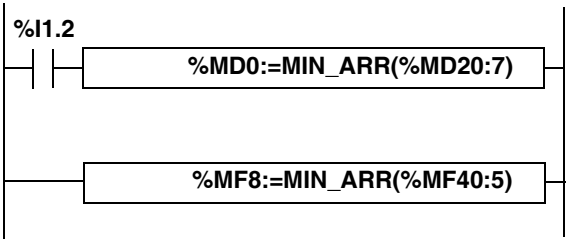
Sono proposte 2 funzioni di ricerca:

- **MAX_ARR**: ricerca del valore massimo in un tabella di parole doppie e virgole mobili
- **MIN_ARR**: ricerca del valore minimo in un tabella di parole doppie e virgole mobili

Il risultato di queste istruzioni è uguale al valore massimo (o minimo) trovato nella tabella.

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %I1.2
[%MD0:=MIN_ARR(%MD20:7)]
%MF8:=MIN_ARR(%MF40:5)
```

Sintassi

Sintassi delle istruzioni di ricerca dei valori massimo e minimo nelle tabelle:

Funzione	Sintassi
MAX_ARR	Res:=Funzione(Tab)
MIN_ARR	

Parametri delle istruzioni di ricerca dei valori massimo e minimo nelle tabelle:

Tipo	Risultato (Ris)	Tabella (Tab)
Tabelle di parole doppie	%MDi	%MDi:L,%KDi:L
Tabelle in virgola mobile	%MFi	%MFi:L,%KFi:L

Numero di occorrenze di un valore in una tabella

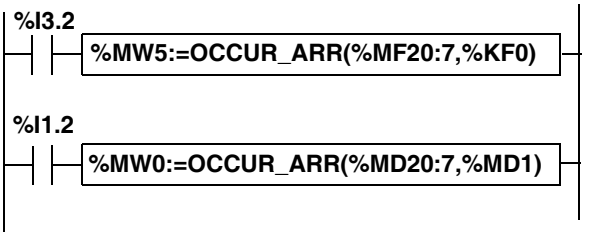
Punti generali

La funzione di ricerca proposta:

- **OCCUR_ARR** : esegue la ricerca in una tabella di parole doppie o di virgole mobili del numero di elementi uguali a un valore dato

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %I3.2
[%MW5:=OCCUR_ARR(%MF20:7,%KF0)]
LD %I1.2
[%MW0:=OCCUR_ARR(%MD20:7,%MD1)
```

Sintassi

Sintassi delle istruzioni di ricerca dei valori massimo e minimo nelle tabelle:

Funzione	Sintassi
OCCUR_ARR	Ris: = Funzione (Tab, Val)

Parametri delle istruzioni di ricerca dei valori massimo e minimo nelle tabelle:

Tipo	Risultato (Ris)	Tabella (Tab)	Valore (Val)
Tabelle di parole doppie	%MWi	%MDi:L,%KDi:L	%MDi,%KDi
Tabelle in virgola mobile	%MFi	%MFi:L,%KFi:L	%MFi,%KFi

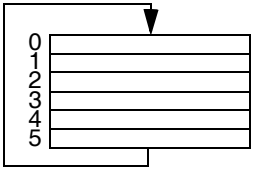
Funzione di scorrimento circolare su una tabella

Punti generali

Sono proposte 2 funzioni di scorrimento ciclico:

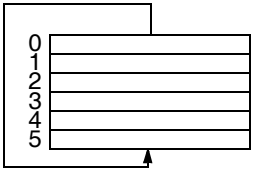
- **ROL_ARR** : esegue lo scorrimento ciclico di n posizioni dall'alto verso il basso di elementi di una tabella a virgola mobile

Illustrazione delle funzioni ROL_ARR



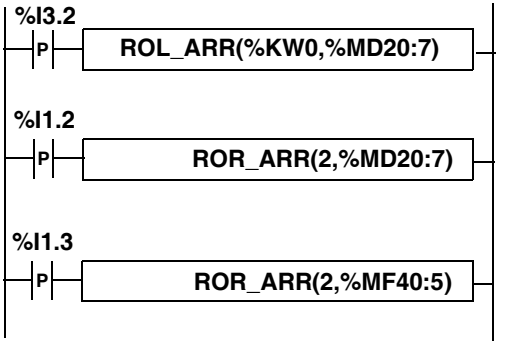
- **ROR_ARR** : esegue lo scorrimento ciclico di n posizioni dall'alto verso il basso di elementi di una tabella a virgola mobile

Illustrazione delle funzioni ROR_ARR



Struttura

Linguaggio Ladder



Linguaggio List

```
LDR %I3.2
[ROL_ARR(%KW0,%MD20:7)]
LDR %I1.2
 [ROR_ARR(2,%MD20:7)]
LDR %I1.3
[ROR_ARR(2,%MF40:5)]
```

Sintassi

Sintassi delle istruzioni di scorrimento circolare sulle tabelle di parole doppie o virgole mobili **ROL_ARR** e **ROR_ARR**

Funzione	Sintassi
ROL_ARR	Funzione(n,Tab)
ROR_ARR	

Parametri delle istruzioni di scorrimento circolare sulle tabelle a virgola mobile: **ROL_ARR** e **ROR_ARR**:

Tipo	Numero di posizioni (n)	Tabella (Tab)
Tabelle in virgola mobile	%MWi, valore immediato	%MFi:L
Tabelle di parole doppie	%MWi, valore immediato	%MDi:L

Nota: se il valore n è negativo o nullo, non viene effettuato nessuno scorrimento.

Funzione di ordinamento sulle tabelle

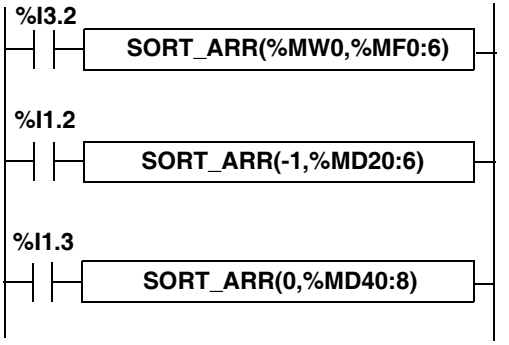
Punti generali

La funzione di ordinamento proposta è la seguente:

- **SORT_ARR** : esegue l'ordinamento in ordine crescente o decrescente degli elementi di una tabella di parole doppie o di virgole mobili e ne memorizza il risultato nella stessa tabella.

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %I3.2
[SORT_ARR(%MW20,%MF0:6)]
LD %I1.2
[SORT_ARR(-1,%MD20:6)]
LD %I1.3
[SORT_ARR(0,%MF40:8)
```

Sintassi

Sintassi delle funzioni di ordinamento sulle tabelle:

Funzione	Sintassi
SORT_ARR	Funzione (senso, Tab)

- il parametro "senso" indica la direzione in cui viene effettuato l'ordinamento: direzione > 0 l'ordinamento è in ordine crescente, direzione < 0 l'ordinamento è in ordine decrescente, direzione = 0 non si esegue alcun ordinamento.
- il risultato (tabella di ordinamento) viene posto nel parametro Tab (tabella da ordinare).

Parametri delle funzioni di ordinamento sulle tabelle:

Tipo	Senso dell'ordinamento	Tabella (Tab)
Tabelle di parole doppie	%MWi, valore immediato	%MDi:L
Tabelle in virgola mobile	%MWi, valore immediato	%MFi:L

Funzione di interpolazione su una tabella a virgola mobile

Panoramica

La funzione **LKUP** permette di interpolare un insieme di dati a virgola mobile X e Y per un dato valore X.

Regola di interpolazione

La funzione LKUP utilizza la regola di interpolazione definita nella seguente equazione:

(equazione 1:) $$Y = Y_i + \left[\frac{(Y_{i+1} - Y_i)}{(X_{i+1} - X_i)} \cdot (X - X_i)\right]$$

per $X_i \le X \le X_{i+1}$, dove $i = 1 \ldots (m-1)$;

ipotizzando che i valori X_i siano disposti in ordine ascendente:

$X_1 \le X_2 \le \ldots X \ldots \le X_{m-1} \le X_m$.

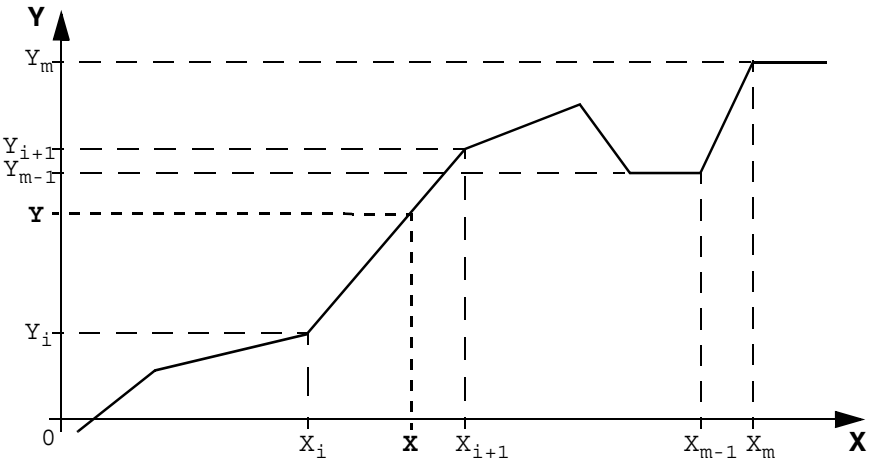
Nota: se due valori Xi consecutivi sono uguali ($X_i=X_{i+1}=X$), l'equazione (1) produce un'eccezione non valida. In questo caso viene usato l'algoritmo seguente al posto dell'equazione (1):

(equazione 2:) $$Y = \left[\frac{(Y_{i+1} - Y_i)}{2}\right]$$

per $X_i = X_{i+1} = X$, dove $i = 1 \ldots (m-1)$.

Rappresentazione grafica della regola di interpolazione lineare

Il grafico seguente mostra la regola di interpolazione lineare descritta sopra:



Sintassi della funzione LKUP

La funzione LKUP utilizza tre operandi, due dei quali sono attributi di funzione, come descritto nella tabella seguente:

Sintassi	**Operando 1 (Op1) Variabile di uscita**	**Operando 2 (Op2) Valore (X) definito dall'utente**	**Operando 3 (Op3) Array di variabili (X_i,Y_i) definito dall'utente**
[Op1: = LKUP(Op2,Op3)]	%MWi	%MF0	Valore intero, %MWi o %KWi

Definizione di Op1

Op1 è la parola di memoria che contiene la variabile di uscita della funzione di interpolazione.
A seconda del valore di Op1, l'utente può sapere se l'interpolazione è stata eseguita correttamente o meno e, in quest'ultimo caso, quali ne sono state cause, come descritto nella tabella seguente:

Op1 (%Mwi)	Descrizione
0	Interpolazione riuscita
1	Errore di interpolazione: Array errato, $X_m < X_{m-1}$
2	Errore di interpolazione: Op2 fuori intervallo, $X < X_1$
4	Errore di interpolazione: Op2 fuori intervallo, $X > X_m$
8	Dimensioni di array di dati non valide: • Op3 è impostato come numero dispari oppure • Op3 < 6.

Nota: Op1 **non** contiene il valore di interpolazione calcolato (Y). Per un valore (X) dato, il risultato dell'interpolazione (Y) è contenuto in %MF2 dell'array Op3 (vedere più avanti *Definizione di Op3*).

Definizione di Op2

Op2 è la variabile a virgola mobile (%MF0 dell'array a virgola mobile Op3) che contiene il valore (X) definito dall'utente per calcolare il valore (Y) interpolato:

- L'intervallo valido per Op2 è il seguente: $X_1 \leq Op2 \leq X_m$.

Definizione di Op3

Op3 determina le dimensioni (Op3 / 2) dell'array a virgola mobile in cui sono memorizzate le coppie di dati (X_i,Y_i).
I dati X_i e Y_i sono memorizzati in oggetti a virgola mobile con indici pari, a partire da %MF4 (si noti che gli oggetti a virgola mobile %MF0 e %MF2 sono riservati rispettivamente per il setpoint dell'utente X e per il valore interpolato Y).

Dato un array di (m) coppie di dati (X_i,Y_i), l'indice superiore (u) dell'array a virgola mobile (%MFu) viene impostando usando le relazioni seguenti:

- *(equazione 3:)* $Op3 = 2 \cdot m$;
- *(equazione 4:)* $u = 2 \cdot (Op3 - 1)$.

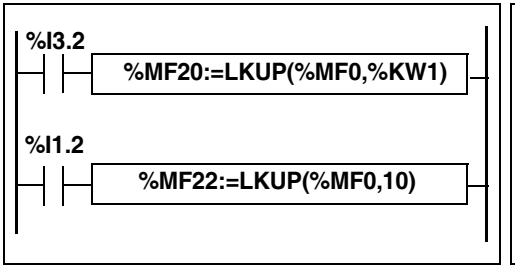
L'array a virgola mobile Op3 (%MFi) ha una struttura simile a quella dell'esempio seguente (dove Op3=8):

(X)		(X_1)		(X_2)		(X_3)	
%MF0		**%MF4**		**%MF8**		**%MF12**	
	%MF2		**%MF6**		**%MF10**		**%MF14**
	(Y)		(Y_1)		(Y_2)		(Y_3)
							(Op3=8)

Nota: come risultato della struttura a virgola mobile dell'array, Op3 deve soddisfare entrambi i seguenti requisiti, altrimenti si produce un errore della funzione LKUP:
- Op3 è un numero pari e
- Op3 ≥ 6 (poiché devono esistere almeno due punti di dati per consentire l'interpolazione lineare).

Struttura

Le operazioni di interpolazione vengono eseguite nel seguente modo:




```
LD      %I3.2
[%MF20:=LKUP(%MF0,%KW1)]

LD      %I1.2
[%MF22:=LKUP(%MF0,10)]
```

Esempio

Di seguito è riportato un esempio di uso della funzione di interpolazione LKUP:

```
[%MW20:=LKUP(%MF0,10)]
```

In questo esempio:

- %MW20 è Op1 (la variabile di uscita).
- %MF0 è il valore (X) definito dall'utente il cui valore (Y) corrispondente deve essere calcolato dall'interpolazione lineare.
- %MF2 memorizza il valore (Y) calcolato risultante dall'interpolazione lineare.
- 10 è Op3 (come risultante dall'*equazione 3* riportata sopra). Determina le dimensioni dell'array a virgola mobile. L'elemento di livello superiore è %MFu, dove u=18 è fornito dall'*equazione 4* riportata sopra.

Quattro 4 coppie di punti di dati sono memorizzate nell'array Op3 [%MF4..%MF18]:

- %MF4 contiene X_1,%MF6 contiene Y_1.
- %MF8 contiene X_2,%MF10 contiene Y_2.
- %MF12 contiene X_3,%MF14 contiene Y_3.
- %MF16 contiene X_4,%MF18 contiene Y_4.

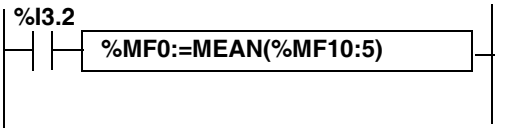
Funzione di calcolo della media dei valori di una tabella a virgola mobile

Punti generali

La funzione **MEAN** consente di calcolare la media dei valori di un numero dato di punti di una tabella a virgola mobile.

Struttura

Linguaggio Ladder



Linguaggio List

```
LD %I3.2
[%MF0:=MEAN(%MF10:5)]
```

Sintassi

Sintassi della funzione di calcolo della media di una tabella a virgola mobile:

Funzione	Sintassi
MEAN	Risultato=Funzione(Op1)

Parametri della funzione di calcolo di un numero dato L di valori di una tabella a virgola mobile:

Operando (Op1)	Risultato (Result)
%MFi:L, %KFi:L	%MFi

Bit di sistema e parole di sistema

18

In breve

Oggetto di questo capitolo

Questo capitolo fornisce una panoramica dei bit e delle parole di sistema che possono essere utilizzati per creare programmi di controllo per i controller Twido.

Contenuto di questo capitolo

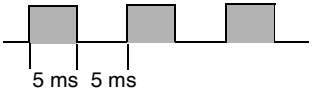
Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Bit di sistema (%S)	604
Parole di sistema (%SW)	614

Bit di sistema (%S)

Introduzione La sezione seguente fornisce informazioni dettagliate sulla funzione dei bit di sistema e sul modo in cui essi vengono controllati.

Descrizione dettagliata La tabella che segue fornisce una panoramica dei bit di sistema e di come vengono controllati:

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S0	Avvio a freddo	Normalmente impostato allo stato 0, questo bit passa a 1 con: • il ripristino dell'alimentazione con perdita dei dati (guasto di una batteria), • il programma utente o l'editor della tabella di animazione, • Pannello visualizzatore. Il bit è impostato a 1 durante la prima scansione. Viene quindi azzerato dal sistema prima della scansione successiva.	0	S o U->S
%S1	Avvio a caldo	Normalmente impostato allo stato 0, questo bit passa a 1 con: • il ripristino dell'alimentazione con backup dei dati, • il programma utente o l'editor della tabella di animazione, • Pannello visualizzatore. Viene azzerato dal sistema alla fine della scansione.	0	S o U->S
%S4 %S5 %S6 %S7	Base tempo: 10 ms Base tempo: 100 ms Base tempo: 1 s Base tempo: 1 min	I cambiamenti di stato di questi bit sono controllati da un orologio interno. Non sono sincronizzati con la scansione del controller. Esempio: %S4  	-	S

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S8	Test di cablaggio	Questo bit, inizialmente a 1, viene utilizzato per il test del cablaggio quando il controller si trova nello stato "non configurato". Per modificare il valore di questo bit, utilizzare i tasti del visualizzatore, quindi cambiare lo stato delle uscite desiderate: ● allo stato 1, corrisponde all'azzeramento delle uscite, ● allo stato 0, permette il test del cablaggio.	1	U
%S9	Azzeramento delle uscite	Normalmente impostato a 0, può essere impostato a 1 dal programma o dal terminale (nell'editor della tabella di animazione): ● Allo stato 1, forza il valore delle uscite a 0 quando il controller è in modalità RUN, ● Allo stato 0, le uscite vengono aggiornate normalmente.	0	U
%S10	Errore di I/O	Normalmente impostato a 1, questo bit è azzerato dal sistema quando viene rilevato un errore di I/O.	1	S
%S11	Superamento del watchdog	Normalmente impostato a 0, questo bit viene impostato a 1 dal sistema quando il tempo di esecuzione del programma (tempo di scansione) supera il tempo di scansione massimo (watchdog del software). Il superamento del watchdog fa passare il controller in modalità HALT.	0	S
%S12	Controller in RUN	Questo bit rispecchia lo stato di esecuzione del controller. Il sistema imposta il bit a 1 quando il controller è in corso di esecuzione. All'arresto, durante l'inizializzazione del sistema o qualsiasi altro stato, il bit viene impostato a 0.	0	S
%S13	Primo ciclo in RUN	Normalmente allo stato 0, questo bit viene impostato a 1 dal sistema durante la prima scansione dopo che il controller è passato in modalità di esecuzione (RUN).	1	S
%S17	Superamento della capacità	Normalmente impostato a 0, viene impostato a 1 dal sistema: ● durante un'operazione di rotazione o di scorrimento. Il sistema imposta l'uscita del bit a 1. Deve essere provato dal programma utente dopo ogni operazione che implica un rischio di overflow, quindi azzerato dall'utente in caso di overflow.	0	S->U

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S9	Azzeramento delle uscite	Normalmente impostato a 0, può essere impostato a 1 dal programma o dal terminale (nell'editor della tabella di animazione): ● Allo stato 1, forza il valore delle uscite a 0 quando il controller è in modalità RUN, ● Allo stato 0, le uscite vengono aggiornate normalmente.	0	U
%S10	Errore di I/O	Normalmente impostato a 1, questo bit è azzerato dal sistema quando viene rilevato un errore di I/O.	1	S
%S11	Superamento del watchdog	Normalmente impostato a 0, questo bit viene impostato a 1 dal sistema quando il tempo di esecuzione del programma (tempo di scansione) supera il tempo di scansione massimo (watchdog del software). Il superamento del watchdog fa passare il controller in modalità HALT.	0	S
%S12	Controller in RUN	Questo bit rispecchia lo stato di esecuzione del controller. Il sistema imposta il bit a 1 quando il controller è in corso di esecuzione. All'arresto, durante l'inizializzazione del sistema o qualsiasi altro stato, il bit viene impostato a 0.	0	S
%S13	Primo ciclo in RUN	Normalmente allo stato 0, questo bit viene impostato a 1 dal sistema durante la prima scansione dopo che il controller è passato in modalità di esecuzione (RUN).	1	S
%S17	Superamento della capacità	Normalmente impostato a 0, viene impostato a 1 dal sistema: ● durante un'operazione di rotazione o di scorrimento. Il sistema imposta l'uscita del bit a 1. Deve essere provato dal programma utente dopo ogni operazione che implica un rischio di overflow, quindi azzerato dall'utente in caso di overflow.	0	S->U

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S18	Superamento o errore aritmetico	Normalmente a 0. Viene impostato a 1 in caso di overflow quando viene eseguita un'operazione a 16 bit, ossia: ● un risultato superiore a + 32.767 o inferiore a – 32.768, in lunghezza semplice, ● un risultato superiore a + 2.147.483.647 o inferiore a – 2.147.483.648, in lunghezza doppia, ● un risultato superiore a + 3.402824E+38 o inferiore a - 3.402824E+38, in virgola mobile, ● una divisione per 0, ● la radice quadrata di un numero negativo, ● una conversione BTI o ITB non significativa: un valore BCD fuori dai limiti. Deve essere provato dal programma utente dopo ogni operazione che implica un rischio di superamento, quindi azzerato dall'utente nel caso in cui si verifichi il superamento.	0	S->U
%S19	Superamento del periodo di scansione (scansione periodica)	Normalmente allo stato 0, questo bit viene impostato a 1 dal sistema in caso di supero del periodo di scansione (tempo di scansione superiore al periodo definito dall'utente nella configurazione o programmato in %SW0). Questo bit viene azzerato dall'utente.	0	S->U
%S20	Superamento dell'indice	Normalmente allo stato 0, viene impostato a 1 quando l'indirizzo dell'oggetto indicizzato diventa inferiore a 0 o superiore della sua dimensione massima. Deve essere provato con il programma utente dopo ogni operazione che implica il rischio di superamento, quindi azzerato nel caso in cui il superamento si verifichi.	0	S->U

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S21	Inizializzazione GRAFCET	Normalmente impostato allo stato 0, questo bit passa a 1 con: ● un riavvio a freddo, %S0=1, ● il programma utente, solo nella parte del programma di preelaborazione, utilizzando un'istruzione Set (S %S21) o una bobina d'eccitazione -(S)- %S21, ● il terminale. Allo stato 1, provoca l'inizializzazione del GRAFCET. Vengono disabilitati i passi attivi e vengono attivati i passi iniziali. Viene successivamente azzerato dal sistema dopo l'inizializzazione del GRAFCET.	0	U->S
%S22	Azzeramento del GRAFCET	Normalmente impostato a 0, può essere impostato a 1 dal programma solo durante la preelaborazione. Allo stato 1 provoca la disabilitazione dei passi attivi dell'intero del GRAFCET. Viene azzerato dal sistema all'inizio dell'esecuzione dell'elaborazione sequenziale.	0	U->S
%S23	Preposizionamento e blocco del GRAFCET	Normalmente impostato a 0, può essere impostato a 1 dal programma solo nel modulo del programma di preelaborazione. Allo stato 1 convalida il preposizionamento del GRAFCET. Il mantenimento di questo bit a 1 determina il blocco del GRAFCET (blocco del grafico). Viene azzerato dal sistema all'avvio dell'esecuzione dell'elaborazione sequenziale per assicurare i cambiamenti del GRAFCET dalla condizione di blocco.	0	U->S
%S24	Pannello visualizzatore	Normalmente impostato a 0, può essere impostato a 1 dall'utente: ● Allo stato 0 il pannello visualizzatore funziona normalmente, ● Allo stato 1 il pannello visualizzatore è bloccato sulla visualizzazione corrente, il lampeggio è disabilitato e i tasti risultano disabilitati.	0	U->S

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S25	Scelta della modalità di visualizzazione sul pannello	È possibile scegliere tra due modalità di visualizzazione sul pannello visualizzatore su due righe: modalità dati e modalità normale. ● Se %S25=0, è abilitata la modalità normale. Sulla prima riga, è possibile scrivere il nome di un oggetto: parola di sistema, parola di memoria, bit di sistema e così via. Sulla seconda riga, è possibile leggerne il valore. ● Se %S25=1, è abilitata la modalità dati. Sulla prima riga, è possibile visualizzare il valore di %SW68. Sulla seconda riga, è possibile visualizzare il valore di %SW69. Quando %S25=1, la tastiera dell'operatore è disabilitata. **Nota:** è necessario disporre del firmware in versione V3.0 o successiva.	0	U
%S26	Scelta del valore con o senza segno sul pannello visualizzatore	È possibile scegliere tra due tipi di valore: con o senza segno. ● Se %S26=0, è abilitata la visualizzazione con segno (da -32768 a 32767). All'inizio di ogni riga sono visualizzati i segni +/-. ● Se %S26=1, è abilitata la visualizzazione senza segno (da 0 a 65535). È possibile utilizzare %S26 solo se %S25=1. **Nota:** è necessario disporre del firmware in versione V3.0 o successiva.	0	U
%S31	Maschera dell'evento	Normalmente a 1. ● Allo stato 0, gli eventi non possono essere eseguiti e vengono posti in attesa. ● Allo stato 1, gli eventi possono essere eseguiti, Questo bit può essere impostato allo stato iniziale 1 dall'utente e dal sistema (riavvio a freddo).	1	U->S
%S38	Autorizzazione degli eventi a posizionarsi in coda	Normalmente a 1. ● Allo stato 0, gli eventi non possono essere posizionati nella coda degli eventi. ● Allo stato 1, gli eventi vengono posizionati nella coda degli eventi non appena vengono rilevati, Questo bit può essere impostato allo stato iniziale 1 dall'utente e dal sistema (riavvio a freddo).	1	U->S

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S39	Stato di saturazione della coda eventi	Normalmente a 0. ● A 0, vengono riportati tutti gli eventi, ● A 1, viene perso almeno un evento. Può essere azzerato dall'utente e dal sistema (durante un avvio a freddo).	0	U->S
%S50	Aggiornamento di data e ora tramite le parole da %SW49 a %SW53	Questo bit, in genere è a 0, ma può essere impostato a 1 o a 0 dal programma o dal pannello visualizzatore. ● Allo stato 0, è possibile leggere data e ora, ● Allo stato 1, è possibile aggiornare data e ora. L'orologio datario interno del controller è aggiornato sul fronte di discesa di %S50.	0	U->S
%S51	Stato dell'orologio Data/ Ora	Questo bit, in genere è a 0, ma può essere impostato a 1 o a 0 dal programma o dal pannello visualizzatore. ● Allo stato 0, è la data e l'ora sono coerenti. ● Allo stato 1 la data e l'ora devono essere inizializzate dall'utente. Se questo bit è impostato a 1, i dati dell'orologio calendario non sono validi. È possibile che la data e l'ora non siano mai state configurate, che il livello di carica della batteria sia basso o che la costante di correzione del controller non sia valida (mai stata configurata, differenza tra la correzione dell'orologio e il valore salvato o il valore è fuori campo). Il passaggio dallo stato 1 allo stato 0 forza la scrittura della costante di correzione sull'orologio in tempo reale.	0	U->S
%S52	Errore RTC	Questo bit gestito dal sistema indica che la correzione dell'orologio in tempo reale non è stata inserita e che data e ora sono errate. ● Allo stato 0, è la data e l'ora sono coerenti. ● Allo stato 1, la data e l'ora devono essere inizializzate.	0	S
%S59	Aggiornamento di data e ora tramite la parola %SW59	Questo bit, in genere è a 0, ma può essere impostato a 1 o a 0 dal programma o dal pannello visualizzatore. ● A 0, la parola di sistema %SW59 non viene gestita, ● Allo stato 1 la data e l'ora vengono incrementate o decrementate in base ai fronti di salita sui bit di controllo impostati in %SW59.	0	U

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S66	Attivazione/ disattivazione BAT LED (solo su controller che supportano una batteria esterna: controller TWDLCA•40DRF).	Questo bit di sistema può essere impostato dall'utente. Consente all'utente di attivare/disattivare il led della batteria (BAT LED): ● Allo stato 0, il BAT LED è attivato (viene azzerato dal sistema all'accensione). ● Allo stato 1, il BAT LED è disattivato (il LED resta spento anche se la carica della batteria esterna è bassa o se non è presente alcuna batteria esterna nello scomparto).	0	S o U->S
%S69	Visualizzatore led di stato (STAT LED) dell'utente	Allo stato 0, il led di stato (STAT LED) è spento. Allo stato 1, il led di stato (STAT LED) è acceso.	0	U
%S75	Stato della batteria esterna (solo su controller che supportano una batteria esterna: controller TWDLCA•40DRF).	Questo bit di sistema viene impostato dal sistema. Indica lo stato della batteria esterna e può essere letto dall'utente: ● Allo stato 0, la batteria funziona normalmente. ● Allo stato 1, la carica della batteria esterna è bassa o non è presente alcuna batteria esterna nello scomparto.	0	S
%S95	Ripristino delle parole di memoria	Questo bit può essere impostato quando le parole di memoria sono state salvate in precedenza nella EEPROM interna. Al completamento, il sistema imposta di nuovo questo bit a 0 e il numero delle parole di memoria ripristinate viene impostato in %SW97	0	U
%S96	Programma di backup corretto	Questo bit può essere letto in qualsiasi momento (dal programma o durante la regolazione), in particolare dopo un avvio a freddo o un riavvio a caldo. ● Allo stato 0, il programma di backup non è valido. ● Allo stato 1, il programma di backup è valido.	0	S
%S97	Salvataggio di %MW corretto	Questo bit può essere letto in qualsiasi momento (dal programma o durante la regolazione), in particolare dopo un avvio a freddo o un riavvio a caldo. ● Allo stato 0, il salvataggio di %MW non è corretto. ● Allo stato 1, il salvataggio di %MW è corretto.	0	S
%S100	Connessione via cavo per comunicazioni TwidoSoft	Indica se il cavo di comunicazione TwidoSoft è collegato. ● Allo stato 1 il cavo di comunicazione TwidoSoft non è collegato oppure TwidoSoft è connesso. ● Allo stato 0, il cavo di collegamento remoto TwidoSoft è collegato.	-	S

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S101	Cambiamento dell'indirizzo di una porta (protocollo Modbus)	Permette di modificare l'indirizzo di una porta tramite le parole di sistema %SW101 (porta 1) e %SW102 (porta 2). Per fare questo, %S101 deve essere impostato a 1. ● Allo stato 0, non è possibile cambiare d'indirizzo. Il valore di %SW101 e di %SW102 corrisponde all'indirizzo corrente della porta, ● Allo stato 1 è possibile cambiare l'indirizzo cambiando i valori di %SW101 (porta 1) e di %SW102 (porta 2). Dopo aver modificato i valori delle parole di sistema, occorre reimpostare %S101 allo stato 0.	0	U
%S103 %S104	Uso del protocollo ASCII	Permette di utilizzare il protocollo ASCII sulla Comm 1 (%S103) o la Comm 2 (%S104). Il protocollo ASCII sarà configurato tramite le parole di sistema %SW103 e %SW105 per la Comm 1 e %SW104 e %SW106 per la Comm 2. ● Allo stato 0, il protocollo utilizzato è quello configurato in TwidoSoft, ● Allo stato 1, si utilizza il protocollo ASCII sulla Comm 1 (%S103) o la Comm 2 (%S104). In questo caso, occorre configurare prima le parole di sistema %SW103 e %SW105 per la Comm 1 e %SW104 e %SW106 per la Comm 2.	0	U
%S110	Scambi in collegamento remoto	Questo bit viene azzerato con il programma o con il terminale. ● Allo stato 1 per un master, tutti gli scambi in collegamento remoto (solo I/O remoti) vengono completati. ● Allo stato 1 per uno slave, lo scambio con il master è terminato.	0	S->U
%S111	Scambio in collegamento remoto singolo	● Allo stato 0 per un master, viene completato uno scambio in collegamento remoto singolo. ● Allo stato 1 per un master, è abilitato uno scambio in collegamento remoto singolo.	0	S
%S112	Connessione in collegamento remoto	● Allo stato 0 per un master, il collegamento remoto è abilitato. ● Allo stato 1 per un master, il collegamento remoto è disabilitato.	0	U

Bit di sistema	Funzione	Descrizione	Stato iniziale	Controllo
%S113	Configurazione/ funzionamento della connessione remota	• Allo stato 0 per un master o uno slave, la configurazione/funzionamento della connessione remota è corretta. • Allo stato 1 per un master, la configurazione/ funzionamento della connessione remota presenta un errore. • Allo stato 1 per uno slave, la configurazione/ funzionamento della connessione remota presenta un errore.	0	S->U
%S118	Errore di I/O remoto	Normalmente impostato a 1, questo bit può essere azzerato quando si rileva un guasto di I/O sul collegamento remoto.	1	S
%S119	Errore di I/O locale	Normalmente impostato a 1, questo bit può essere azzerato quando si rileva un guasto di I/O sul collegamento remoto. %SW118 determina la natura dell'errore. Reimpostato a 1 quando viene eliminato l'errore.	1	S

Descrizione delle abbreviazioni della tabella

Tabella delle abbreviazioni:

Abbreviazione	Descrizione
S	Controllato dal sistema
U	Controllato dall'utente
U->S	Impostato a 1 dall'utente, azzerato dal sistema
S->U	Impostato a 1 dal sistema, azzerato dall'utente

Parole di sistema (%SW)

Introduzione

La sezione seguente fornisce informazioni dettagliate sulla funzione delle parole di sistema e sul relativo modo di regolazione.

Descrizione dettagliata

La tabella seguente fornisce informazioni dettagliate sulla funzione delle parole di sistema e sul modo in cui vengono controllate.

Parole di sistema	Funzione	Descrizione	Controllo
%SW0	Periodo di scansione del controller (task periodico)	Modifica il periodo di scansione del controller definito al momento della configurazione mediante il programma utente e l'editor delle tabelle di animazione.	U
%SW1	Salva il valore di un evento periodico	Modifica il tempo di ciclo [5-255 ms] di un evento periodico senza perdere il valore Periodo salvato nella casella degli eventi periodici della finestra Modalità di scansione. Consente di recuperare il valore del periodo salvato nella casella degli eventi periodici: ● in caso di avvio a freddo o ● nel caso il valore scritto in %SW1 sia esterno all'intervallo [5-255]. Il valore di %SW1 può essere modificato al termine di un ciclo, nel programma o nella tabella di animazione, senza dover interrompere il programma. Durante l'esecuzione del programma è possibile osservare correttamente i tempi di ciclo.	U
%SW6	Stato del controller	Stato del controller: 0 = NO CONFIG 2 = STOP 3 = RUN 4 = HALT	S

Parole di sistema	Funzione	Descrizione	Controllo

%SW7	Stato del controller	• Bit [0]: backup/ripristino in corso:
		  • a 1 se backup/ripristino in corso,
		  • a 0 se backup/ripristino terminato o non attivo.
		• Bit [1]: configurazione corretta del controller:
		  • a 1 se la configurazione è corretta.
		• Bit [3..2] Bit di stato della EEPROM:
		  • 00 = Nessuna cartuccia
		  • 01 = Cartuccia EPROM da 32 KB
		  • 10 = Cartuccia EPROM da 64 KB
		  • 11 = Riservato per uso futuro
		• Bit [4]: [4] Applicazione nella RAM diversa da quella nella EEPROM:
		  • a 1 se applicazione nella RAM diversa da quella nella EEPROM.
		• Bit [5]: Applicazione nella RAM diversa da quella nella cartuccia:
		  • a 1 se applicazione nella RAM diversa da quella nella cartuccia.
		• Bit [6] Non utilizzato (stato 0)
		• Bit [7]: Controller riservato:
		  • a 1 se riservato.
		• Bit [8]: Applicazione in modalità scrittura:
		  • a 1 se l'applicazione è protetta.
		• Bit [9] Non utilizzato (stato 0)
		• Bit [10]: Seconda porta seriale installata:
		  • a 1 se è installata.
		• Bit [11]: Secondo tipo di porta seriale: (0 = EIA RS-232, 1 = EIA RS-485):
		  • a 0 = EIA RS-232
		  • a 1 = EIA RS-485
		• Bit [12]: Applicazione valida nella RAM:
		  • a 1 se l'applicazione è valida.
		• Bit [13] Applicazione valida nella cartuccia:
		  • a 1 se l'applicazione è valida.
		• Bit [14] Applicazione valida nella RAM:
		  • a 1 se l'applicazione è valida.
		• Bit [15]: Pronto per l'esecuzione:
		  • a 1 se pronto per l'esecuzione.
			S

%SW11	Valore del watchdog del software	Contiene il valore massimo del watchdog. Il valore (da 10 a 500 ms) è definito dalla configurazione.	U

Parole di sistema	Funzione	Descrizione	Controllo
%SW14	Versione commerciale, Vxx.yy	Ad esempio, se %SW14=0232: ● 8 MSB=02 in formato esadecimale, quindi xx=2 in decimale ● 8 LSB=32 in formato esadecimale, quindi yy=50 in decimale Di conseguenza, la versione commerciale è V2.50. **Nota:** è necessario disporre del firmware in versione 2.5 o successiva.	S
%SW15	Patch firmware, Pzz	Ad esempio, se %SW15=0005: ● 8 MSB non è utilizzato ● 8 LSB=05 in formato esadecimale, quindi zz=5 in decimale Di conseguenza, la patch firmware è P05. **Nota:** è necessario disporre del firmware in versione 2.5 o successiva.	S
%SW16	Versione firmware, Vxx.yy	Ad esempio, se %SW16=0232: ● 8 MSB=02 in formato esadecimale, xx=2 in decimale ● 8 LSB=32 in formato esadecimale, quindi yy=50 in decimale Di conseguenza, la versione firmware è V2.50. **Nota:** è necessario disporre del firmware in versione 2.5 o successiva.	S
%SW17	Stato predefinito per operazione in virgola mobile	Se viene individuato un errore in un'operazione aritmetica in virgola mobile, il bit %S18 viene impostato a 1 e lo stato predefinito di %SW17 viene aggiornato secondo la codifica seguente: ● Bit [0]: operazione non valida, il risultato non è un numero (1.#NAN o -1.#NAN), ● Bit 1: riservato, ● Bit 2: divisione per 0, il risultato è infinito (-1.#INF o 1.#INF), ● Bit 3: risultato superiore in valore assoluto a +3.402824e+38, il risultato è infinito (-1.#INF o 1.#INF),	S e U
%SW18-%SW19	Contatore timer assoluto da 100 ms	Il contatore utilizza due parole: ● **%SW18** rappresenta la parola meno significativa, ● **%SW19** rappresenta la parola più significativa.	S e U
da %SW20 a %SW27	Fornisce lo stato dei moduli slave CANopen con indirizzo nodo da 1 a 16.	Per ulteriori informazioni, vedere *Parole di sistema specifiche riservate slave CANopen , p. 275*.	S
%SW30	Durata dell'ultima scansione	Mostra il tempo di esecuzione dell'ultimo ciclo di scansione del controller (in ms). **Nota:** questo tempo corrisponde al tempo trascorso tra l'inizio (acquisizione degli ingressi) e la fine (aggiornamento delle uscite) di un ciclo di scansione.	S

Parole di sistema	Funzione	Descrizione	Controllo

%SW31	Tempo di scansione massimo	Mostra il tempo di esecuzione del ciclo di scansione più lungo del controller dopo l'ultimo avvio a freddo (in minuti).
		Note:
		• questo tempo corrisponde al tempo trascorso tra l'inizio (acquisizione degli ingressi) e la fine (aggiornamento delle uscite) di un ciclo di scansione.
		• Tenere presente che per consentire il corretto rilevamento di un segnale di impulso quando è selezionata l'opzione di ingresso di latch, l'ampiezza dell'impulso (T_{ON}) e il periodo ciclico ($T_{impulso}$) devono soddisfare le due condizioni seguenti:
		  • $T_{ON} \geq 1$ ms
		  • Il periodo ciclico del segnale di ingresso deve seguire la regola di campionamento di Nyquist-Shannon secondo cui il periodo ciclico ($T_{impulso}$) del segnale di ingresso deve essere almeno doppio del tempo di scansione massimo del programma (%SW31):
		  $T_{impulso} \geq 2$ x %SW31.
		  Nota: se questa condizione non viene rispettata, si potrebbero perdere alcuni impulsi.
			S

%SW32	Tempo di scansione minimo	Mostra il tempo di esecuzione del ciclo di scansione più corto del controller dopo l'ultimo avvio a freddo (in minuti). **Nota:** questo tempo corrisponde al tempo trascorso tra l'inizio (acquisizione degli ingressi) e la fine (aggiornamento delle uscite) di un ciclo di scansione.	S
%SW48	Numero di eventi	Visualizza il numero di eventi eseguiti dall'ultimo avvio a freddo. Sono conteggiati tutti gli eventi esclusi quelli periodici. **Nota:** impostato a 0 (dopo caricamento dell'applicazione e avvio a freddo), incrementa per ogni esecuzione di evento.	S

Parole di sistema	Funzione	Descrizione		Controllo
%SW49 %SW50 %SW51 %SW52 %SW53	Orologio in tempo reale (RTC)	Funzione di programmazione (RTC): le parole che contengono i valori di data e ora correnti (in BCD):		S e U
		%SW49	xN giorno della settimana (N=1 per lunedì)	
		%SW50	00SS secondi	
		%SW51	HHMM ore e minuti	
		%SW52	MMGG mese e giorno	
		%SW53	SSAA secolo e anno	
		Queste parole sono controllate dal sistema quando il bit **%S50** è a 0. Queste parole possono essere scritte dal programma utente o dal terminale quando il bit **%S50** è impostato a 1. Sul fronte di discesa di **%S50**, l'orologio in tempo reale interno del controller viene aggiornato dai valori scritti in queste parole.		
%SW54 %SW55 %SW56 %SW57	Data e ora dell'ultimo arresto	Le parole di sistema che contengono la data e l'ora dell'ultima interruzione di alimentazione o del controller (in BCD):		S
		%SW54	SS secondi	
		%SW55	HHMM ore e minuti	
		%SW56	MMGG mese e giorno	
		%SW57	SSAA secolo e anno	
%SW58	Codice dell'ultima interruzione	Visualizza il codice indicando la causa dell'ultima interruzione:		S
		1 =	Fronte d'ingresso Run/Stop	
		2 =	Interruzione per guasto software (overflow della scansione del controller)	
		3 =	Comando Stop	
		4 =	Interruzione alimentazione	
		5 =	Interruzione per guasto hardware	

Parola di sistema	Funzione	Descrizione			Controllo
%SW59	Impostare la data corrente	Imposta la data corrente. Contiene due set di 8 bit per regolare la data corrente. L'operazione viene sempre eseguita sul fronte di salita del bit. Questa parola viene abilitata dal bit **%S59**.			U
		Incremento	**Decremento**	**Parametro**	
		bit 0	bit 8	Giorno della settimana	
		bit 1	bit 9	Secondi	
		bit 2	bit 10	Minuti	
		bit 3	bit 11	Ore	
		bit 4	bit 12	Giorni	
		bit 5	bit 13	Mese	
		bit 6	bit 14	Anni	
		bit 7	bit 15	Secoli	
%SW60	Correzione RTC	Valore di correzione dell'orologio in tempo reale			U
%SW63	Codice di errore blocco EXCH1	Codice di errore EXCH1: 0 - Operazione riuscita 1 - Numero di byte da emettere troppo grande (> 250) 2 - Tabella di emissione troppo piccola 3 - Tabella di parole troppo piccola 4 - Overflow tabella di ricezione 5 - Timeout scaduto 6 - Invio 7 - Comando errato nella tabella 8 - Porta selezionata non configurata/disponibile 9 – Errore di ricezione 10 - Impossibile utilizzare %KW se in ricezione 11 - Offset di trasmissione più grande della tabella di trasmissione 12 - Offset di ricezione più grande della tabella di ricezione 13 - Interruzione dell'elaborazione di EXCH da controller			S
%SW64	Codice di errore blocco EXCH2	Codice d'errore EXCH2: Vedere %SW63.			S

Parola di sistema	Funzione	Descrizione	Controllo
%SW65	Codice di errore blocco EXCH3	Il codice di errore EXCH3 è implementato solo sui controller Twido TWDLCAE40DRF che supportano Ethernet. 1-4, 6-13: Vedere %SW63. Si noti che il codice di errore 5 non è valido e viene sostituito dai codici di errore 109 e 122 specifici di Ethernet descritti di seguito. Di seguito sono indicati i codici di errore specifici di Ethernet: 101 - Indirizzo IP non esistente 102 - Connessione TCP interrotta 103 - Nessun socket disponibile (tutti i canali di connessione sono occupati) 104 - Interruzione di rete 105 - Impossibile raggiungere la rete 106 - Interruzione di rete al reset 107 - Connessione terminata dal dispositivo peer 108 - Connessione ripristinata dal dispositivo peer 109 - Timeout di connessione scaduto 110 - Tentativo di connessione rifiutato 111 - Host spento 120 - Indice sconosciuto (il dispositivo remoto non è indicizzato nella tabella di configurazione) 121 - Irreversibile (MAC, Chip, IP doppio)122 - Timeout di ricezione scaduto dopo l'invio dei dati 123 - Inizializzazione Ethernet in corso	S
%SW67	Funzione e tipo di controller	Contiene le seguenti informazioni: ● Bit del tipo di controller [0 -11] ● 8B0 = TWDLC•A10DRF ● 8B1 = TWDLC•A16DRF ● 8B2 = TWDLMDA20DUK/DTK ● 8B3 = TWDLC•A24DRF ● 8B4 = TWDLMDA40DUK/DTK ● 8B6 = TWDLMDA20DRT ● 8B8 = TWDLCAA40DRF ● 8B9 = TWDLCAE40DRF ● Bit 12,13,14,15 non utilizzati = 0	S

Parole di sistema	Funzione	Descrizione	Controllo
%SW68 e %SW69	Elementi da visualizzare contemporaneamente sul pannello a due righe	Se %S25=1, la modalità di visualizzazione dei dati è abilitata. La tastiera dell'operatore è disabilitata. %SW68 e %SW69 possono essere visualizzati contemporaneamente sul pannello a due righe: ● %SW68 valore della prima riga, ● %SW69 valore della seconda riga, **Nota:** è necessario disporre del firmware in versione V3.0 o successiva.	U
%SW73 e %SW74	Stato del sistema AS-Interface	● Bit [0]: a 1 se la configurazione è corretta. ● Bit [1]: a 1 se è attivato lo scambio dati. ● Bit [2]: a 1 se il modulo è in modalità Offline. ● Bit [3]: a 1 se l'istruzione ASI_CMD è terminata. ● Bit [4]: a 1 se è in corso errore dell'istruzione ASI_CMD.	S e U
da %SW76 a %SW79	Contatori indietro 1-4	Queste quattro parole fungono da timer da 1 ms. Vengono decrementate singolarmente dal sistema ogni ms se hanno un valore positivo. Ciò fornisce 4 contatori indietro che contano in ms, il che corrisponde a un campo operativo compreso tra 1 e 32767 ms. L'impostazione del bit 15 a 1 può interrompere il conteggio indietro.	S e U
%SW80	Stato degli I/O della base	Bit [0] Canali in funzionamento normale (per tutti i canali). Bit [1] Modulo in inizializzazione (oppure informazioni di inizializzazione per tutti i canali). Bit [2] Guasto hardware (guasto alimentazione esterna, comune a tutti i canali) Bit [3] Errore di configurazione del modulo Bit [4] Conversione della canale d'ingresso dei dati 0 in corso Bit [5] Conversione del canale d'ingresso dei dati 1 in corso Bit [6] Ingresso termocoppia canale 0 non configurato Bit [7] Ingresso termocoppia canale 1 non configurato Bit [8] Non utilizzato Bit [9] Non utilizzato Bit [10] Ingresso analogico dati canale 0 al di sopra dell'intervallo Bit [11] Ingresso analogico dati canale 1 al di sopra dell'intervallo Bit [12] Collegamento errato (canale dei dati d'ingresso analogico 0 inferiore all'intervallo corrente, loop di corrente aperto) Bit [13] Collegamento errato (canale dei dati d'ingresso analogico 1 inferiore all'intervallo corrente, loop di corrente aperto) Bit [14] Non utilizzato Bit [15] Canale di uscita non disponibile	S

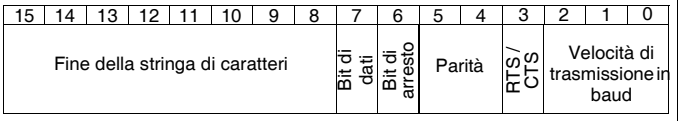
Parole di sistema	Funzione	Descrizione	Controllo
%SW81	• Stato modulo di espansione I/O 1: Definizioni come %SW80 • Stato del modulo master CANopen all'indirizzo di espansione 1: • Bit [0] Stato configurazione (1 = configurazione corretta; 0 = errore di configurazione) Bit [1] Stato operativo (1 = scambio PDO attivato; 0 = scambio PDO disattivato) Bit [2] Stato iniziale (1 = stato iniziale attivato; 0 = stato iniziale disattivato) Bit [3] Istruzione CAN_CMD completa (1 = completa; 0 = in corso) Bit [4] Errore istruzione CAN_CMD (1 = errore; 0 = corretta) Bit [5] Errore d'inizializzazione (1 = errore; 0 = corretta) Bit [6] Perdita del messaggio, errore dell'alimentazione (1 = errore; 0 = corretta)		S
%SW82	Stato modulo di espansione I/O 2: Definizioni come %SW80 Stato del modulo master CANopen all'indirizzo di espansione 2: Definizioni come %SW81		S
%SW83	Stato modulo di espansione I/O 3: Definizioni come %SW80 Stato del modulo master CANopen all'indirizzo di espansione 3: Definizioni come %SW81		S
%SW84	Stato modulo di espansione I/O 4: Definizioni come %SW80 Stato del modulo master CANopen all'indirizzo di espansione 4: Definizioni come %SW81		S
%SW85	Stato modulo di espansione I/O 5: Definizioni come %SW80 Stato del modulo master CANopen all'indirizzo di espansione 5: Definizioni come %SW81		S
%SW86	Stato modulo di espansione I/O 6: Definizioni come %SW80 Stato del modulo master CANopen all'indirizzo di espansione 6: Definizioni come %SW81		S
%SW87	Stato modulo di espansione I/O 7: Definizioni come %SW80 Stato del modulo master CANopen all'indirizzo di espansione 7: Definizioni come %SW81		S
%SW94	Firma dell'applicazione	In caso di modifica dell'applicazione, in termini di configurazione o di dati di programmazione, la firma (somma di tutte le checksum) viene modificata di conseguenza. Se %SW94=91F3 in formato esadecimale, la firma dell'applicazione è 91F3 in esadecimale. **Nota:** è necessario disporre del firmware in versione V2.5 o successiva.	S

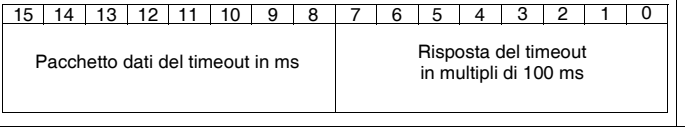
Parole di sistema	Funzione	Descrizione	Controllo

%SW96	Comando e/o diagnostica per funzione di salvataggio/ ripristino del programma applicativo e %MW.	• Bit [0]: indica che occorre eseguire il backup nella EEPROM delle parole di memoria %MW:
		  • a 1 se è richiesto backup,
		  • a 0 se il backup in corso non è terminato.
		• Bit [1]: questo bit viene impostato dal firmware per indicare quando il backup è terminato:
		  • a 1 se il backup è terminato,
		  • a 0 se viene richiesto un nuovo backup.
		• Bit [2]: errore di backup, vedere i bit 8, 9, 10 e 14 per maggiori informazioni:
		  • a 1 se si è verificato un errore,
		  • a 0 se viene richiesto un nuovo backup.
		• Bit [6]: a 1 se il controller contiene un'applicazione valida.
		• Bit [8]: indica che il numero di %MWs specificato in %SW97 è maggiore del numero di %MWs configurate nell'applicazione:
		  • a 1 se viene individuato l'errore,
		• Bit [9]: indica che il numero di %MWs specificato in %SW97 è maggiore del numero massimo di %MWs definibili da qualsiasi applicazione in TwidoSoft.
		  • a 1 se viene individuato l'errore,
		• Bit [10]: differenza tra RAM interna e EEPROM interna (1 = sì).
		  • a 1 in presenza di differenza.
		• Bit [14]: indica se si è verificato un errore di scrittura nella EEPROM:
		  • a 1 se viene individuato un errore,
			S e U

%SW97	Comando o diagnostica per la funzione di salvataggio/ ripristino	Quando si salvano le parole di memoria, questo valore rappresenta il numero fisico %MW da salvare nella EEPROM interna. Quando si ripristinano le parole di memoria, il valore viene aggiornato con il numero di parole di memoria ripristinato nella RAM. Per l'operazione di salvataggio, se questo numero è 0, le parole di memoria non vengono memorizzate. L'utente deve impostare la logica del programma. In caso contrario, sarà azzerata nell'applicazione del controller ad eccezione del seguente caso: All'avvio a freddo, questa parola è impostata a -1 se la Flash EEPROM interna non ha parole di memoria **%MW** salvate. Nel caso di un avvio a freddo in cui la Flash EEPROM interna contiene un elenco di parole di memoria **%MW**, il valore del numero di parole di memoria salvate nel file deve essere scritto nella parola di sistema **%SW97**.	S e U

Parole di sistema	Funzione	Descrizione	Controllo
%SW101 %SW102	Valore dell'indirizzo Modbus della porta	Quando il bit %S101 è impostato a 1, è possibile modificare l'indirizzo Modbus della porta 1 o della porta 2. L'indirizzo della porta 1 è %SW101 e quello della porta 2 è %SW102.	S

%SW103
%SW104
	Configurazione per l'utilizzo del protocollo ASCII	Quando il bit %S103 (Comm 1) o il bit %S104 (Comm 2) è impostato a 1, si utilizza il protocollo ASCII. La parola di sistema %SW103 (Comm 1) o %SW104 (Comm 2) deve essere parametrizzata in funzione degli elementi dati qui di seguito:
		
		
		• Velocità in baud:
		  • 0: 1200 baud
		  • 1: 2400 baud
		  • 2: 4800 baud
		  • 3: 9600 baud
		  • 4: 19200 baud
		  • 5: 38400 baud.
		• RTS/CTS:
		  • 0: disattivato
		  • 1: attivato.
		• Parità:
		  • 00: nessuna
		  • 10: dispari
		  • 11: pari
		• Bit di stop:
		  • 0: 1 bit di stop
		  • 1: 2 bit di stop
		• Bit d'errore:
		  • 0: 7 bit di dati
		  • 1: 8 bit di dati.
			S

%SW105 %SW106	Configurazione per l'utilizzo del protocollo ASCII	Quando il bit %S103 (Comm 1) o il bit %S104 (Comm 2) è impostato a 1, si utilizza il protocollo ASCII. La parola di sistema %SW105 (Comm 1) o %SW106 (Comm 2) deve essere parametrizzata in funzione degli elementi dati qui di seguito:  	S

Parole di sistema	Funzione	Descrizione	Controllo
%SW111	Stato del collegamento remoto	Indicazione: il bit 0 corrisponde al controller remoto 1, il bit 1 al controller remoto 2, e così via. Bit da [0] a [6]: ● allo stato 0 = controller remoto 1-7 assente ● allo stato 1 = controller remoto 1-7 presente Bit da [8] a [14]: ● allo stato 0 = I/O remoto rilevato al controller remoto 1-7 ● allo stato 1 = controller peer rilevato nel controller remoto 1-7	S
%SW112	Codice di errore per configurazione/ funzionamento collegamento remoto	00: operazioni riuscite 01: scadenza timeout (slave) 02: rilevato errore di checksum (slave) 03: mancata corrispondenza della configurazione (slave) Impostato dal sistema a 1, deve essere azzerato dall'utente.	S
%SW113	Configurazione collegamento remoto	Indicazione: il bit 0 corrisponde al controller remoto 1, il bit 1 al controller remoto 2, e così via. Bit da [0] a [6]: ● a 0 = controller remoto 1-7 non configurato ● a 1 = controller remoto 1-7 configurato Bit da [8] a [14]: ● a 0 = I/O remoto configurato come controller remoto 1-7 ● a 1 = controller peer configurato come controller remoto 1-7	S
%SW114	Abilitazione dei blocchi di programmazione	Abilita o disabilita il funzionamento dei blocchi di programmazione da parte del programma utente o del pannello visualizzatore. Bit 0: 1 = abilita blocco di programmazione 0 ... Bit 15: 1 = abilita blocco di programmazione 15 Inizialmente tutti i blocchi di programmazione sono abilitati. Se i blocchi di programmazione sono configurati il valore predefinito è FFFF Se nessun blocco di programmazione è configurato il valore predefinito è 0.	S e U
%SW118	Parola di stato del controller della base	Mostra gli errori rilevati sul controller master. Bit 9: 0= Errore esterno o di comunicazione Bit 12: 0 = RTC non installato Bit 13: 0 = Errore di configurazione (estensione di I/O configurata ma assente o difettosa). Tutti gli altri bit di questa parola sono impostati a 1 e riservati. Per un controller che non presenta guasti, il valore di questa parola è FFFFh.	S

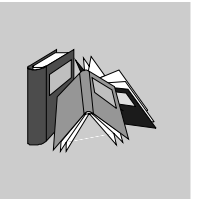
Parole di sistema	Funzione	Descrizione	Controllo
%SW120	Condizione del modulo I/O di espansione	Un bit per modulo. Indirizzo 0 = Bit 0 1 = Condizione non corretta 0 = OK	S

Descrizione delle abbreviazioni della tabella

Tabella delle abbreviazioni:

Abbreviazione	Descrizione
S	Controllato dal sistema
U	Controllato dall'utente

Glossario



!

% Prefisso che identifica gli indirizzi della memoria interna del controller utilizzati per memorizzare il valore di variabili di programma, costanti, I/O e così via.

A

Analizza programma Comando che compila un programma e verifica se vi sono errori. Rileva errori di sintassi e di struttura, simboli privi degli indirizzi corrispondenti, risorse non disponibili utilizzate dal programma e se il programma supera la memoria disponibile del controller. Gli errori sono visualizzati nel Visualizzatore degli errori di programma.

Applicazione Un'applicazione TwidoSoft è composta da un programma, dai dati di configurazione, dai simboli e dalla documentazione.

ASCII (American Standard Code for Information Interchange) Protocollo di comunicazione per rappresentare caratteri alfanumerici, incluse lettere, numeri e alcuni caratteri grafici e di controllo.

Autocaricamento Questa funzione è sempre abilitata e permette di trasferire automaticamente un'applicazione da una cartuccia di backup alla RAM del controller in caso di perdita o danneggiamento dell'applicazione stessa. All'accensione, il controller confronta l'applicazione presente nella propria RAM con quella memorizzata nella cartuccia di memoria opzionale di backup, se installata. Se rileva differenze, la copia che si trova nella cartuccia di backup viene copiata nel controller e nella EEPROM interna. Se la cartuccia di backup non è installata, nel controller viene copiata l'applicazione che si trova nella EEPROM interna.

Avvio o riavvio a freddo Avvio determinato dal controller in cui tutti i dati sono inizializzati ai valori predefiniti e il programma riprende dall'inizio con tutte le variabili cancellate. Tutte le impostazioni software e hardware sono inizializzate. Un riavvio a freddo può essere provocato dal caricamento di una nuova applicazione nella RAM del controller. Tutti i controller senza batterie di backup si accendono sempre con un riavvio a freddo

B

Backup Comando che copia l'applicazione che si trova nella RAM del controller sia nella EEPROM interna del controller sia nella cartuccia di memoria opzionale di backup, se installata.

Blocchi di programmazione Blocco funzione che consente di programmare le funzioni Data e Ora per controllare gli eventi. Richiede l'opzione RTC (Orologio in tempo reale).

Blocco funzione Unità di programma costituita da ingressi e variabili organizzati in modo da calcolare i valori delle uscite in base a una funzione definita, ad esempio un timer o un contatore.

Bobina Elemento del linguaggio Ladder che rappresenta un'uscita dal controller.

BootP Un protocollo (Bootstrap Protocol) basato su UDP/IP che consente ad un host, all'avvio, di autoconfigurarsi dinamicamente senza l'intervento dell'utente. BootP è in grado di notificare a un host l'assegnazione del suo indirizzo IP.

Browser dell'applicazione Finestra specializzata di TwidoSoft che visualizza graficamente la vista gerarchica di un'applicazione. Consente di configurare e visualizzare agevolmente un'applicazione.

Bus di estensione I moduli di estensione degli I/O si collegano al controller di base tramite questo bus.

C

CAN — **Controller Area Network**: bus di campo sviluppato originariamente per applicazioni nel campo delle automobili, utilizzato ormai in molti settori, dal comparto industriale al terziario.

Cancellazione — Questo comando consente di eliminare l'applicazione nel controller e presenta due opzioni:
- Per eliminare il contenuto della memoria RAM del controller, della memoria EEPROM interna del controller e della cartuccia opzionale di backup installata.
- Per cancellare solo il contenuto della cartuccia di backup.

Cartuccia di memoria — Cartucce di memoria opzionali di backup utilizzabili per eseguire il backup e il ripristino di un'applicazione (programma e dati di configurazione). Sono disponibili due dimensioni: 32 e 64 KB.

CiA — **CAN in Automation**: organizzazione internazionale di utenti e fabbricanti di prodotti CAN.

Client — Processo di computer che richiede assistenza da altri processi di computer.

COB — **Oggetto di comunicazione**: unità di trasporto sul bus CAN. Un COB è identificato da un identificatore univoco, codificato a 11 bit, [0, 2047] e contiene massimo 8 byte di dati. La priorità di trasmissione del COB è mostrata dall'identificatore: meno forte è l'identificatore, minore priorità avrà il relativo COB.

Collegamento remoto — Bus master/slave ad alta velocità progettato per comunicare una piccola quantità di dati tra un controller master e fino a sette controller remoti (slave). È possibile configurare due tipi di controller remoti per trasferire i dati a un controller master: un controller peer che può trasferire i dati dell'applicazione e un controller remoto I/O che può trasferire i dati di I/O. Una rete di collegamento remoto può includere entrambi i tipi di controller.

Commenti — I commenti sono costituiti da un testo immesso per documentare lo scopo di un programma. Per i programmi Ladder, è possibile immettere fino a tre righe di testo nell'Intestazione della rete per descriverne lo scopo. Ogni riga può includere fino a 64 caratteri. Per i programmi List, è possibile immettere il testo su n righe di programma non numerate. I commenti devono essere inseriti tra parentesi e asterischi nel modo seguente: (*QUESTI SONO I COMMENTI*).

Contatore — Blocco funzione utilizzato per contare gli eventi avanti o indietro.

Contatori molto veloci (VFC) — Blocco funzione che consente un conteggio più veloce rispetto ai blocchi funzione Contatori e Contatori veloci. Un contatore molto veloce (VFC) può raggiungere la velocità di 20 KHz.

Contatori veloci (FC)	Blocco funzione che consente un conteggio avanti o indietro più veloce rispetto al blocco funzione Contatori. Un contatore veloce (FC) può raggiungere la velocità di 5 KHz.
Contatto	Elemento del linguaggio Ladder che rappresenta un ingresso per il controller.
Controller	Controller programmabile Twido. Sono disponibili due tipi di controller: Compatto e Modulare.
Controller a tamburo	Blocco funzione che funziona in modo analogo a un controller a tamburo elettromeccanico in cui il passo varia in relazione a eventi esterni.
Controller Compatto	Tipo di controller Twido con configurazione semplice e compatta e con espansione limitata. L'altro tipo di controller Twido è Modulare.
Controller master	Controller Twido configurato come principale su una rete di collegamento remoto.
Controller Modulare	Tipo di controller Twido che offre una configurazione flessibile con capacità di espansione. L'altro tipo di controller Twido è Compatto.
Controller peer (di estensione)	Controller Twido configurato come slave in una rete con collegamento in remoto. È possibile eseguire un'applicazione nella memoria del controller peer. Il programma potrà avere accesso ai dati di I/O locali e delle estensioni ma non è possibile trasferire i dati degli I/O al controller master. Il programma eseguito dal controller peer passa le informazioni al controller master tramite parole di rete (%INW e %QNW).
Controller programmabile	Controller Twido. Sono disponibili due tipi di controller: Compatto e Modulare.
Controller remoto	Controller Twido configurato per comunicare con un controller principale su una rete di collegamento remoto.
Convalida automatica linea	Durante l'inserimento o la modifica delle istruzioni List, questa impostazione opzionale consente di convalidare le righe di programma man mano che vengono immesse dopo aver verificato che non vi siano errori e simboli non risolti. È necessario correggere ogni elemento prima di uscire dalla riga. Per selezionare questa opzione, utilizzare la finestra di dialogo Preferenze.
Costanti	Valore configurato che non può essere modificato dal programma in esecuzione.

E

Editor delle tabelle di animazione	Finestra specializzata dell'applicazione TwidoSoft per la visualizzazione e la creazione di tabelle di animazione.
Editor di configurazione	Finestra specializzata di TwidoSoft che consente di gestire la configurazione hardware e software.
Editor Ladder	Finestra specializzata di TwidoSoft che consente di modificare un programma Ladder.
Editor List	Semplice editor di programmi che consente di creare e modificare un programma List.
EDS	**Electronic Data Sheet**: file di descrizione per ogni dispositivo CAN, fornito dal produttore.
EEPROM	Electrically Erasable Programmable Read-Only Memory. Twido ha una EEPROM interna e una cartuccia di memoria EEPROM esterna opzionale.
Executive Loader	Applicazione Windows a 32 bit utilizzata per scaricare un nuovo programma Firmware Executive in un controller Twido.

F

FIFO	First In, First Out; Primo in entrata, primo in uscita. Blocco funzione utilizzato per le operazioni di coda.
File dell'applicazione	Le applicazioni Twido sono memorizzate come file di tipo .twd.
Firmware Executive	Firmware Executive è il sistema operativo che esegue le applicazioni e gestisce il funzionamento del controller.
Forzatura	Impostazione volontaria degli ingressi e delle uscite del controller ai valori 0 o 1 anche se i valori effettivi sono diversi. Utilizzata per il debugging durante l'animazione di un programma.

Frame Un gruppo di bit che forma un blocco digitale di informazioni. I frame contengono informazioni o dati di controllo della rete. La dimensione e la composizione di un frame è determinata dalla tecnologia di rete utilizzata.

Funzionamento in linea Modalità operativa di TwidoSoft quando si collega un PC al controller e l'applicazione nella memoria del PC è uguale a quella presente nella memoria del controller. Durante il funzionamento consente di eseguire il debugging dell'applicazione.

Funzionamento Offline (non in linea) Una modalità operativa di TwidoSoft quando al controller non è collegato alcun PC e l'applicazione nella memoria del PC non è uguale a quella presente nella memoria del controller. Le applicazioni vengono create e sviluppate in modalità Offline.

Funzioni Data/ Orologio Permette il controllo degli eventi in base al mese, al giorno del mese e all'ora del giorno. Vedere "Blocchi orologio-datario".

G

Gateway Un dispositivo che collega le reti con architetture di rete dissimili e che opera a livello applicazione. Questo termine può riferirsi a un router.

Gateway predefinito L'indirizzo IP della rete o dell'host verso il quale sono inviati tutti i pacchetti indirizzati verso una rete o un host sconosciuto. Il gateway predefinito è tipicamente un router o un altro dispositivo di instradamento.

Gestore risorse Componente di TwidoSoft che controlla il fabbisogno di memoria di un'applicazione durante la programmazione e la configurazione collegando i riferimenti agli oggetti software creati da un'applicazione. Un oggetto si considera referenziato da un'applicazione se viene utilizzato come operando in una istruzione List o in una rete Ladder. Visualizza le informazioni di stato riguardanti la percentuale di memoria totale utilizzata e genera avvertenze nel caso la memoria stia per esaurirsi. Vedere Indicatore dell'uso di memoria

Grafcet Il linguaggio Grafcet permette di rappresentare graficamente e in maniera strutturata il funzionamento di un sistema di automazione sequenziale.
Si tratta di un metodo analitico che divide i sistemi di controllo sequenziali in una serie di passi a cui sono associate azioni, transizioni e condizioni.

H

Host Un nodo su una rete.

Hub Un dispositivo che permette di collegare una serie di moduli flessibili e centralizzati per creare una rete.

I

Indicatore dell'uso di memoria Parte della barra di stato della finestra principale di TwidoSoft che visualizza la percentuale della memoria totale del controller utilizzata da un'applicazione. Visualizza un avviso quando la memoria sta per esaurirsi.

Indirizzi Registri interni del controller utilizzati per memorizzare i valori di variabili di programma, costanti, I/O e così via. Gli indirizzi sono identificati dal prefisso del simbolo di percentuale (%). %I0.1 ad esempio specifica un indirizzo della memoria RAM del controller contenente il valore del canale di ingresso 1.

Indirizzo IP Indirizzo del protocollo Internet. Un indirizzo a 32 bit assegnato all'host utilizzando i protocolli TCP/IP.

Indirizzo MAC Indirizzo di controllo d'accesso media L'indirizzo hardware di un dispositivo Un indirizzo MAC è assegnato in fabbrica a un modulo TCP/IP Ethernet.

Ingresso Latch Gli impulsi in ingresso sono acquisiti e registrati per essere esaminati successivamente da parte dell'applicazione.

Inizializza Comando che imposta tutti i valori dei dati agli stati iniziali. Il controller deve trovarsi in modalità Stop o Error.

Internet Le reti di comunicazione dei computer basate su protocollo TCP/IP per un'interconnessione globale.

Intestazione della rete Pannello visualizzato direttamente sopra una rete Ladder al fine di documentarne lo scopo.

IP Internet Protocol. Un protocollo comune a livello di rete. IP è spesso usato con il protocollo TCP.

Istanza Oggetto univoco di un programma che appartiene a un tipo specifico di blocco funzione. Nel formato timer %TMi, ad esempio, i è un numero che rappresenta l'istanza.

Istruzioni reversibili Metodo di programmazione che consente di visualizzare le istruzioni alternativamente come istruzioni List o come reti Ladder.

L

LIFO Last In, First Out; Ultimo in entrata, primo in uscita. Blocco funzione utilizzato per le operazioni di stack.

Linguaggio Ladder Un programma scritto in linguaggio Ladder è costituito da una rappresentazione grafica delle istruzioni di un programma del controller che include simboli per i contatti, le bobine e i blocchi di una serie di reti eseguite sequenzialmente dal controller.

Linguaggio Lista d'istruzioni (List) Un programma scritto in linguaggio Lista d'istruzioni (IL, Instruction List) è costituito da una serie di istruzioni eseguite sequenzialmente dal controller. Ogni istruzione è composta da un numero di riga, un codice istruzione e un operando.

M

MBAP Modbus Application Protocol (Procollo per applicazioni Modbus)

Modalità di scansione Specifica le modalità di scansione di un programma da parte del controller. Sono possibili due tipi di modalità di scansione: Normale (ciclica), in cui il controller esegue continuamente la scansione oppure Periodica, in cui il controller esegue la scansione per una durata selezionata (intervallo da 2 a 150 msec) prima di avviare la scansione successiva.

Modbus Protocollo di comunicazione master-slave che consente a un singolo master di richiedere risposte dagli slave.

Moduli di espansione degli I/O Moduli di estensione degli I/O opzionali disponibili per aggiungere punti di I/O a un controller Twido. Non tutti i modelli di controller consentono l'estensione.

N

Nodo Un dispositivo indirizzabile su una rete di comunicazione.

O

Operando Numero, indirizzo o simbolo che rappresenta un valore manipolabile da un programma tramite un'istruzione.

Operatore Simbolo o codice che specifica l'operazione eseguita da un'istruzione.

Orologio in tempo reale Opzione che mantiene l'ora anche quando il controller non è alimentato per un periodo di tempo limitato.

P

Pacchetto L'unità dati inviata su una rete.

Passo Un passo Grafcet indica lo stato di funzionamento sequenziale del controller.

PC Personal Computer.

PLS Generazione di impulsi. Blocco funzione che genera un'onda quadra con un ciclo duty di 50 % On e 50% Off.

Potenziometro analogico Una tensione che può essere regolata e convertita in un valore digitale utilizzabile da un'applicazione.

Preferenze Finestra di dialogo con opzioni selezionabili per impostare gli editor di programma List e Ladder.

Protezione Indica due diversi tipi di protezione delle applicazioni: protezione tramite password per il controllo dell'accesso e protezione dell'applicazione del controller che impedisce tutte le operazioni di lettura e scrittura del programma applicativo.

Protocollo Descrive i formati dei messaggi e imposta un insieme di regole utilizzate da due o più dispositivi per comunicare utilizzando tali formati.

PWM Pulse Width Modulation (Impulso a modulazione d'ampiezza). Blocco funzione che genera un'onda rettangolare con ciclo duty variabile impostabile da un programma.

R

RAM Random Access Memory. Le applicazioni Twido sono scaricate nella RAM interna volatile per essere eseguite.

Registri Registri speciali interni al controller dedicati ai blocchi funzione LIFO/FIFO.

Rete Dispositivi interconnessi che condividono un percorso dati e un protocollo comune di comunicazione.

Rete Una rete viene creata tra due barre dei potenziali in una griglia ed è costituita da un gruppo di elementi grafici collegati tra loro da collegamenti orizzontali o verticali. Le dimensioni massime di una rete sono sette righe e undici colonne.

Rete Ladder Visualizza le parti di un programma List non reversibili al linguaggio Ladder.

Riavvio a caldo Accensione determinata dal controller dopo una perdita di alimentazione senza modifica dell'applicazione. Il controller torna allo stato precedente alla perdita di alimentazione e completa la scansione in corso. Tutti i dati dell'applicazione vengono mantenuti. Questa funzione è disponibile solo sui controller modulari.

Riferimenti incrociati Generazione di un elenco di operandi, simboli, numeri di riga o di rete e operatori utilizzati in un'applicazione per semplificare la creazione e la gestione delle applicazioni.

Righe di commento Nei programmi List, è possibile immettere i commenti su righe separate dalla istruzioni. Le righe di commento non sono numerate e devono essere inserite tra parentesi e asterischi nel modo seguente: (*QUESTI SONO I COMMENTI*).

Router Un dispositivo che permette di collegare due o più sezioni di una rete e consente lo scambio di dati tra le sezioni di rete. Un router esamina ogni pacchetto di dati che riceve e decide se bloccarlo o di trasmetterlo sulla rete. Il router prova sempre a inviare i pacchetti dati su una rete attraverso il percorso più veloce.

RTC Vedere Orologio in tempo reale (da Real-Time Clock).

RTU Remote Terminal Unit. Protocollo basato su otto bit che consente la comunicazione tra un controller e un PC.

Run Comando che avvia l'esecuzione di un programma applicativo da parte del controller.

S

Scansione Il controller esegue la scansione o l'analisi di un programma, svolgendo essenzialmente tre funzioni base: prima di tutto, legge gli ingressi e registra i valori corrispondenti nella memoria. Esegue quindi il programma applicativo, una istruzione alla volta e memorizza i risultati. Utilizza infine i risultati per aggiornare le uscite.

Server Un processo del computer che fornisce servizi ai clienti. Questo termine può anche riferirsi al processo del computer sul quale è basato il servizio.

Simbolo Un simbolo è una stringa di massimo 32 caratteri alfanumerici in cui il primo carattere è alfabetico. Consente di personalizzare un oggetto del controller per agevolare la manutenzione dell'applicazione.

Simbolo non risolto Simbolo privo dell'indirizzo di una variabile.

Stati operativi Indica lo stato operativo di TwidoSoft. Visualizzato sulla barra di stato. Esistono quattro stati operativi: Initial (iniziale), Offline (non in linea), Online (in linea) e Monitor (monitoraggio).

Stato in linea Stato operativo di TwidoSoft che viene visualizzato sulla barra di stato quando si collega un PC al controller.

Stato iniziale Stato operativo di TwidoSoft che viene visualizzato sulla barra di stato quando TwidoSoft viene avviato o quando non è aperta alcuna applicazione.

Stato monitor Stato operativo di TwidoSoft che viene visualizzato sulla barra di stato quando si collega un PC a un controller in modalità di non scrittura.

Stato Offline (non in linea) Stato operativo di TwidoSoft che viene visualizzato sulla barra di stato quando non si collega un PC al controller.

Stop Comando che interrompe l'esecuzione di un programma applicativo da parte del controller.

Subnet Una rete fisica o logica all'interno di una rete IP, che condivide un indirizzo di rete con altre porzioni della rete.

Subnet mask Una maschera di bit utilizzata per identificare o determinare quali bit in un indirizzo IP corrispondono all'indirizzo di rete e quali bit corrispondono alla parte subnet dell'indirizzo. La subnet mask è l'indirizzo di rete più i bit riservati per l'identificazione della sotto rete.

Switch Un dispositivo di rete che permette di collegare due o più segmenti di rete, separati, consentendo il passaggio del traffico di rete tra di loro. Uno switch o commutatore determina se un frame deve essere bloccato o trasmesso in base al suo indirizzo di destinazione.

T

Tabella dei simboli Tabella dei simboli utilizzata in un'applicazione, visualizzata nell'Editor dei simboli.

Tabella di animazione Tabella creata in un editor di linguaggio o in una schermata operativa. Quando si collega un PC al controller, la tabella visualizza le variabili del controller e consente di forzarne i valori durante il debugging. Può essere salvata come un file separato con estensione .tat.

TCP Transmission Control Protocol (protocollo di controllo della trasmissione).

TCP/IP] Una suite di un protocolli composta da un protocollo di controllo della trasmissione e dal protocollo Internet, ossia la suite di protocolli sulla quale è basato Internet.

Timer Blocco funzione che consente di selezionare una durata di tempo per controllare un evento.

Tipi di frame Due tipi di frame comuni sono Ethernet II e IEEE 802.3.

Twido Linea di controller Schneider Electric che include due tipi di controller (Compatto e Modulare), moduli di espansione per aggiungere punti di I/O e opzioni quali l'Orologio in tempo reale, le comunicazioni, il pannello visualizzatore e le cartucce di memoria di backup.

TwidoSoft Software di sviluppo grafico per ambiente Windows a 32 bit per la configurazione e la programmazione dei controller Twido.

U

UDP Un protocollo di comunicazione (User Datagram Protocol) che fa parte del pacchetto TCP/IP utilizzato dalle applicazioni per trasferire i diagrammi di dati. UDP è anche la parte del TCP/IP che gestisce gli indirizzi delle porte.

Uscita riflessa In modalità di conteggio, il valore corrente del contatore molto veloce (%VFC.V) è misurato in rapporto alle soglie configurate, al fine di determinare lo stato di queste uscite dedicate.

Uscite di soglia Bobine controllate direttamente dal contatore molto veloce (%VFC) in base alle impostazioni stabilite nella configurazione.

V

Variabile Unità di memoria indirizzabile e modificabile da un programma.

Variabile di dati Vedere Variabile.

Visualizzatore degli errori di programma Finestra specializzata di TwidoSoft che consente di visualizzare gli errori di programma e le avvertenze.

Visualizzatore dei riferimenti incrociati Finestra specializzata dell'applicazione TwidoSoft per la visualizzazione dei riferimenti incrociati.

Indice analitico



Symbols

-, 573
%Ci, 405
%DR, 460
%FC, 465
%INW, 43
%MSG, 480
%PLS, 457
%PWM, 454
%QNW, 43
%S, 604
%S0, 604
%S1, 604
%S10, 605, 606
%S100, 611
%S101, 612
%S103, 612
%S104, 612
%S11, 605, 606
%S110, 612
%S111, 612
%S112, 612
%S113, 613
%S118, 613
%S119, 613
%S12, 605, 606
%S13, 605, 606
%S17, 605, 606
%S18, 607
%S19, 607
%S20, 607
%S21, 73, 608
%S22, 73, 608
%S23, 73, 608
%S24, 608
%S25, 609
%S26, 609
%S31, 609
%S38, 609
%S39, 610
%S4, 604
%S5, 604
%S50, 610
%S51, 610
%S52, 610
%S59, 610
%S6, 604
%S66, 611
%S69, 611
%S7, 604
%S75, 611
%S8, 605
%S9, 605, 606
%S95, 611
%S96, 611
%S97, 611
%SBR, 410
%SCi, 412
%SW, 614
%SW0, 614
%SW1, 614
%SW101, 624
%SW102, 624
%SW103, 624

%SW104, 624
%SW105, 624
%SW106, 624
%SW11, 615
%SW111, 625
%SW112, 625
%SW113, 625
%SW114, 625
%SW118, 625
%SW120, 626
%SW14, 616
%SW15, 616
%SW16, 616
%SW17, 616
%SW18, 616
%SW19, 616
%SW20..%SW27, 275, 616
%SW30, 616
%SW31, 617
%SW32, 617
%SW48, 617
%SW49, 618
%SW50, 618
%SW51, 618
%SW52, 618
%SW53, 618
%SW54, 618
%SW55, 618
%SW56, 618
%SW57, 618
%SW58, 618
%SW59, 619
%SW6, 614
%SW60, 619
%SW63, 619
%SW64, 619
%SW65, 620
%SW67, 620
%SW68, 621
%SW69, 621
%SW7, 615
%SW73, 621
%SW74, 621
%SW76, 621
%SW77, 621
%SW78, 621
%SW79, 621
%SW80, 621
%SW81..%SW87, 274, 622
%SW94, 622
%SW96, 623
%SW97, 623
%TM, 402
%VFC, 468
*, 573
+, 573
/, 573

A

ABS, 573
Accesso al debug
    PID, 543
Accesso alla configurazione
    PID, 526
Accumulatore, 352
ACOS, 576
Addizione, 424
Analogico, canale, 192
Analogico, modulo
    funzionamento, 194
AND, istruzioni, 384
ASCII
    comunicazione, 88
    comunicazioni, 120
    configurazione della porta, 123
    configurazione hardware, 120
    configurazione software, 122
ASIN, 576
ATAN, 576
Avvio, 246
Avvio a freddo, 78
Azione derivata, 566
Azione integrale, 565
Azione proporzionale, 564
Azione, area, 330

B

Backup e ripristino
cartuccia di backup da 32 K, 58
cartuccia di memoria estesa da 64 K, 61
senza cartuccia, 56
struttura della memoria, 54
Bit di memoria, 27
Bit di scorrimento, registro, 410
Bit di sistema, 604
Bit Run/Stop, 75
BLK, 344
Blocchi
nei linguaggi Ladder, 332
Blocchi funzione
blocchi orologio-datario, 485
contatori, 405
Controller a tamburo, 460
elemento grafico, 337
funzione contatore passo passo (%SCi), 412
presentazione dei blocchi funzione di base, 393
programmatore a tamburo, 463
programmazione dei blocchi funzione standard, 395
PWM, 454
registri, 448
Timer, 397
Blocchi funzione avanzati
oggetti bit e parola, 444
principi di programmazione, 446
Blocchi funzione di base, 393
Blocco funzione %MSG3
Istruzioni, 185
Blocco funzione contatore molto veloce (%VFC), 468
Blocco funzione contatore veloce, 465
Blocco funzione del controller a tamburo, 460
Blocco funzione di scambio, 480
Bobine, 332
elementi grafici, 336
Booleane, istruzioni
Informazioni sul formato usato nel manuale, 378
Booleano, accumulatore, 352
BootP, 166
Bus AS-Interface V2
acquisizione della nuova configurazione, 227
configurazione software, 213
debug del bus, 224
descrizione funzionale generale, 207
diagnostica degli slave, 221
indirizzamento automatico di uno slave, 229
indirizzamento degli I/O, 232
inserimento slave, 230
modalità di funzionamento, 238
modifica dell'indirizzo di uno slave, 222
presentazione, 206
principio di implementazione software, 210
programmazione e diagnostica del bus AS-Interface, 234
scambi espliciti, 234
scambi impliciti, 233
schermata di configurazione, 211
schermata di debug, 219
slave in errore, 231
trasferimento dell'immagine di uno slave, 225
Bus CANopen
metodologia di configurazione, 258
Bus di campo CANopen
programmazione e diagnostica del bus di campo CANopen, 274
scambi espliciti, 274
Scambi impliciti, 273

C

Calcolo, 424
CAN, linea bus, 243
CAN_CMD, 277
CAN-high, 243
CAN-low, 243
CANopen
descrizione, 243
protocollo, 243
Caratteristiche del PID, 523

Ciclo del task, 71
Client/server TCP, 157
Coda d'attesa, 448
Collegamento ASCII
  esempio, 129
Collegamento del cavo di connessione, 90
Collegamento Modbus
  Esempio 1, 144
  esempio 2, 147
Collegamento remoto
  accesso ai dati I/O remoti, 113
  comunicazione, 88
  comunicazioni, 108
  configurazione del controller master, 111
  Configurazione del controller remoto, 111
  Configurazione hardware, 109
  Configurazione software, 111
  esempio, 117
  sincronizzazione della scansione del controller remoto, 112
Commenti delle righe List, 346
Comunicazione
  Modbus, 88
Comunicazione via modem, 90
Comunicazioni
  ASCII, 120
  collegamento remoto, 108
  Modbus, 132
Configurazione
  PID, 526
  porta Modbus, 136
  porta per protocollo ASCII, 123
  tabella di trasmissione e ricezione per ASCII, 124
Configurazione TCP/IP, 170
Confronto, blocchi, 334
Confronto, blocco
  elemento grafico, 337
Connessioni, gestione, 182
Contatori, 405
  Programmazione e configurazione, 408
Contatti, 332
  elemento grafico, 335
Controllo del tempo di scansione, 71
Convalida degli oggetti, 26
Correzione RTC, 484
COS, 576

D

Debug
  PID, 543
Decremento, 424
DEG_TO_RAD, 578
Diagrammi Ladder
  elementi grafici, 335
DINT_TO_REAL, 580
Divisione, 424
Documentazione del programma, 346

E

Elaborazione digitale
  panoramica, 416
Elementi di collegamento
  elementi grafici, 335
Elementi grafici
  diagrammi Ladder, 335
END, istruzioni, 435
END_BLK, 344
EQUAL_ARR, 586
Errore, 426
Esempio
  Contatore avanti/indietro, 409
Ethernet
  configurazione TCP/IP, 170
  connessione di rete, 164
  gestione delle connessioni, 182
EXCH, 479
EXCH, istruzione, 479
EXCH3, 185
  codice di errore, 187
EXP, 573
EXPT, 573

F

Fattore di correzione del pannello visualizzatore, 324

FIFO
funzionamento, 450
introduzione, 448
FIND_, 588
Funzione contatore passo passo, 412
Funzione, blocchi
nella griglia di programmazione, 333
registro bit di scorrimento (%SBR), 410
timer, 402
Funzioni dell'orologio
impostazione della data e dell'ora, 490
Panoramica, 484
Timbro data e ora, 488
Funzioni orologio
blocchi orologio-datario, 485

G

Generazione d'impulsi, 457
Grafcet
azioni associate, 369
elaborazione sequenziale, 367
esempi, 363
istruzioni, 362
pre-elaborazione, 366

I

I/O
indirizzamento, 40
ID unità, 180
Incremento, 424
Indirizzamento
indicizzato, 50
Indirizzamento degli I/O, 40
Indirizzamento dei moduli di I/O analogici, 195
Indirizzamento diretto, 50
Indirizzo del gateway, 165
Indirizzo IP, 165
BootP, 166
indirizzo IP predefinito, 166
Indirizzo MAC, 166
Inizializzazione degli oggetti, 80
INT_TO_REAL, 580
Interruzione dell'alimentazione, 74
IP contrassegnato, 175
Istruzioni
AND, 384
aritmetiche, 424
di confronto, 422
di conversione, 431
JMP, 438
Load, 380
logiche, 428
NOT, 390
RET, 439
SR, 439
XOR, 388
istruzioni
END, 435
NOP, 437
Istruzioni aritmetiche, 424
Istruzioni booleane, 376
assegnazione, 382
OR, 386
Istruzioni d'assegnazione, 382
Istruzioni di assegnazione
numeriche, 417
Istruzioni di confronto, 422
Istruzioni di conversione, 431
Istruzioni di conversione tra parole semplici e doppie, 433
Istruzioni di salto, 438
Istruzioni di scorrimento, 429
Istruzioni di subroutine, 439
Istruzioni digitali
di scorrimento, 429
Istruzioni logiche, 428
Istruzioni numeriche
assegnazione, 417

J

JMP, 438

L

Ladder, linguaggi
blocchi, 332
Introduzione, 328
OPEN e SHORT, 338
principi di programmazione, 330
Ladder, programma
inversione in List, 343
Ladder, reti, 329
LAN ACT, 184
LAN ST, 184
LD, 380
LDF, 377, 380
LDN, 380
LDR, 376, 380
Life guarding, 252
Life time, 252
LIFO
introduzione, 448
Operazione, 449
linea RS-485 EIA), 135
Linguaggi di programmazione
Panoramica, 21
Linguaggio List
panoramica, 350
List, istruzioni, 353
Livello fisico, 243
linea bus CAN, 243
LKUP, 596
LN, 573
LOG, 573

M

Master CANopen
indirizzamento PDO, 273
MAX_ARR, 590
MEAN, 601
Memoria
cartuccia da 32 K, 58
cartuccia da 64 K, 61
senza cartuccia, 56
struttura, 54
Metodi Grafcet, 72
MIN_ARR, 590
Modalità
Operativa, 248
preoperativa, 248
Modalità operative, 72
Modbus
client/server TCP, 157
comunicazioni, 132
configurazione della porta, 136
configurazione hardware, 133
configurazione software, 136
master, 88
Messaggeria TCP Modbus, 185
richieste standard, 150
slave, 88
Modbus TCP/IP
dispositivi remoti, 179
Modulazione d'ampiezza d'impulso, 454
Moduli analogici
configurazione degli I/O, 196
indirizzamento, 195
Modulo analogico
esempio, 203
Moltiplicazione, 424
MPP, 358
MPS, 358
MRD, 358

N

Node guarding, 252
Non condizionali, reti, 345
NOP, 437
NOP, istruzione, 437
NOT, istruzione, 390
Note per la programmazione, 339

O

OCCUR_ARR, 591
Oggetti
blocchi funzione, 44
oggetti bit, 27
parola, 29
parola doppia, 32
strutturati, 46
virgola mobile, 32

Oggetti a parola doppia
panoramica, 32
Oggetti a virgola mobile
indirizzamento, 38
panoramica, 32
Oggetti bit, 444
Indirizzamento, 36
presentazione, 27
Oggetti parola, 444
Indirizzamento, 37
panoramica, 29
Oggetti parola doppia, 45
Oggetti parole doppie
indirizzamento, 39
OPEN, 338
Operandi, 352
Operate, blocchi, 334
elemento grafico, 337
OR esclusivo, istruzioni, 388
OR, istruzione, 386
OUT_BLK, 344
Overflow, 426
indice, 51
Overflow indice, 51

P

Pannello visualizzatore
correzione dell'orologio-datario, 324
ID e stati del controller, 313
impostazioni delle porte seriali, 322
orologio in tempo reale, 323
panoramica, 310
variabili e oggetti di sistema, 315
Panoramica
PID, 519
Panoramica sulla comunicazione, 88
Parametri, 398
Parametri di controllo
ASCII, 124
Parentesi
annidamento, 357
modificatori, 357
utilizzo nei programmi, 356
parole di memoria, 29
Parole di sistema, 614
PID
configurazione, 526
debug, 543
Panoramica, 519
scheda Animazione, 545
scheda AT, 535
scheda generale, 528
scheda Ingresso, 531
scheda PID, 533
scheda Traccia, 547
Scheda Uscita, 540
Pin di uscita
connettore femmina del cavo di comunicazione, 92
connettore maschio del cavo di comunicazione, 92
Polarizzazione (esterna, 135
Potenziometro, 190
Principi di programmazione, 446
Programmatori a tamburo
programmazione e configurazione, 463
Programmazione
documentazione del programma, 346
Programmazione non reversibile, 446
Programmazione reversibile, 446
Programmazione, griglia, 330
Protocolli, 88
Protocollo
Modbus TCP/IP, 89

R

RAD_TO_DEG, 578
Radice quadrata, 424
REAL_TO_DINT, 580
REAL_TO_INT, 580
Registri
FIFO, 450
LIFO, 449
programmazione e configurazione, 451
Regolazione a loop aperto, 562
Regolazione a loop chiuso, 561
Resto, 424
RET, 439
Rete
Indirizzamento, 43

Rete Ladder/List, 345
Rete, intestazione, 331
  commenti, 347
Reti
  Non condizionali, 345
Reversibilità
  indicazioni, 344
  introduzione, 343
Riavvio a caldo, 76
Ricezione di messaggi, 479
Rilevamento fronte
  di discesa, 377
  di salita, 376
Ripristino dell'alimentazione, 74
ROL_ARR, 592
ROR_ARR, 592

S

Scansione
  ciclica, 66
  periodica, 68
Scheda Animazione
  PID, 545
Scheda AT
  PID, 535
Scheda Generale
  PID, 528
Scheda Ingresso
  PID, 531
Scheda PID
  PID, 533
Scheda Traccia
  PID, 547
Scheda Uscita
  PID, 540
SHORT, 338
Simbolizzazione, 52
SIN, 576
SORT_ARR, 594
Sottrazione, 424
SQRT, 573
SR, 439
Stack, 448
Stack, istruzioni, 358
Stringhe di bit, 46
Subnet mask, 165
SUM_ARR, 584

T

Tabella di controllo
  Modbus, 138
Tabelle di oggetti, 46
TAN, 576
Task di evento
  gestione degli eventi, 84
  presentazione, 82
  sorgenti diverse degli eventi, 83
TCP/IP
  protocollo, 89
Tempo di scansione, 71
Test, area, 330
Timeout (Ethernet), 177
Timer, 398
  base tempo di 1 ms, 403
  introduzione, 397
  programmazione e configurazione, 402
  Tipo TOF, 399
  Tipo TON, 400
  TP, tipo, 401
TOF, timer, 399
TON, timer, 400
TP, tipo di timer, 401
Trasmissione di messaggi, 479
TRUNC, 573
TwidoSoft
  introduzione, 20

V

valore assoluto, 424

W

Watchdog software, 71

X

XOR, 388