

SIEMENS

SIMATIC

S7 Sistema di automazione S7-1200

Manuale di sistema

Prefazione

Presentazione del prodotto

1

Montaggio

2

Concetti base sui PLC

3

Configurazione dei dispositivi

4

Concetti di programmazione

5

Istruzioni di programmazione

6

PROFINET

7

Comunicazioni punto a punto
(PtP)

8

Tool online e di diagnostica

9

Dati tecnici

A

Calcolo del budget di
potenza

B

Numeri di ordinazione

C

Avvertenze di legge

Concetto di segnaletica di avvertimento

Questo manuale contiene delle norme di sicurezza che devono essere rispettate per salvaguardare l'incolumità personale e per evitare danni materiali. Le indicazioni da rispettare per garantire la sicurezza personale sono evidenziate da un simbolo a forma di triangolo mentre quelle per evitare danni materiali non sono precedute dal triangolo. Gli avvisi di pericolo sono rappresentati come segue e segnalano in ordine decrescente i diversi livelli di rischio.

PERICOLO

questo simbolo indica che la mancata osservanza delle opportune misure di sicurezza **provoca** la morte o gravi lesioni fisiche.

AVVERTENZA

il simbolo indica che la mancata osservanza delle relative misure di sicurezza **può causare** la morte o gravi lesioni fisiche.

CAUTELA

con il triangolo di pericolo indica che la mancata osservanza delle relative misure di sicurezza può causare lesioni fisiche non gravi.

CAUTELA

senza triangolo di pericolo indica che la mancata osservanza delle relative misure di sicurezza può causare danni materiali.

ATTENZIONE

indica che, se non vengono rispettate le relative misure di sicurezza, possono subentrare condizioni o conseguenze indesiderate.

Nel caso in cui ci siano più livelli di rischio l'avviso di pericolo segnala sempre quello più elevato. Se in un avviso di pericolo si richiama l'attenzione con il triangolo sul rischio di lesioni alle persone, può anche essere contemporaneamente segnalato il rischio di possibili danni materiali.

Personale qualificato

Il prodotto/sistema oggetto di questa documentazione può essere adoperato solo da **personale qualificato** per il rispettivo compito assegnato nel rispetto della documentazione relativa al compito, specialmente delle avvertenze di sicurezza e delle precauzioni in essa contenute. Il personale qualificato, in virtù della sua formazione ed esperienza, è in grado di riconoscere i rischi legati all'impiego di questi prodotti/sistemi e di evitare possibili pericoli.

Uso conforme alle prescrizioni di prodotti Siemens

Si prega di tener presente quanto segue:

AVVERTENZA

I prodotti Siemens devono essere utilizzati solo per i casi d'impiego previsti nel catalogo e nella rispettiva documentazione tecnica. Qualora vengano impiegati prodotti o componenti di terzi, questi devono essere consigliati oppure approvati da Siemens. Il funzionamento corretto e sicuro dei prodotti presuppone un trasporto, un magazzinaggio, un'installazione, un montaggio, una messa in servizio, un utilizzo e una manutenzione appropriati e a regola d'arte. Devono essere rispettate le condizioni ambientali consentite. Devono essere osservate le avvertenze contenute nella rispettiva documentazione.

Marchio di prodotto

Tutti i nomi di prodotto contrassegnati con ® sono marchi registrati della Siemens AG. Gli altri nomi di prodotto citati in questo manuale possono essere dei marchi il cui utilizzo da parte di terzi per i propri scopi può violare i diritti dei proprietari.

Esclusione di responsabilità

Abbiamo controllato che il contenuto di questa documentazione corrisponda all'hardware e al software descritti. Non potendo comunque escludere eventuali differenze, non possiamo garantire una concordanza perfetta. Il contenuto di questa documentazione viene tuttavia verificato periodicamente e le eventuali correzioni o modifiche vengono inserite nelle successive edizioni.

Prefazione

Scopo del manuale

La serie S71200 è una linea di controllori a logica programmabile (PLC) in grado di controllare un'ampia varietà di applicazioni. La compattezza del design, il costo contenuto e l'esteso set di istruzioni fanno dell'S71200 la soluzione ottimale per numerose applicazioni industriali. Inoltre i modelli di S7-1200 e il tool di programmazione su base Windows garantiscono la flessibilità necessaria per affrontare e risolvere i più svariati problemi di automazione.

Il presente manuale contiene informazioni sull'installazione e la programmazione dei PLC S7-1200 e si rivolge a tecnici, programmatori, installatori ed elettricisti che dispongono di conoscenze generiche sui controllori a logica programmabile.

Nozioni di base richieste

Per poter comprendere il contenuto del presente manuale è necessario avere una conoscenza generale dei principi dell'automazione e dei controllori a logica programmabile.

Oggetto del manuale

Il presente manuale ha come oggetto il software STEP 7 Basic V10.5 e la serie di prodotti S71200. L'elenco completo dei prodotti della serie S7-1200 descritti nel presente manuale è riportato nel capitolo Dati tecnici (Pagina 295).

Certificazione, marchio CE, C-Tick e altre norme

Per maggiori informazioni sull'argomento consultare i dati tecnici (Pagina 295).

Servizio clienti e assistenza tecnica

Oltre alla documentazione Siemens fornisce assistenza tecnica alla pagina Internet:

<http://www.siemens.com/automation/support-request>

Per ricevere assistenza su eventuali problemi tecnici, richiedere informazioni sui corsi di formazione e ordinare i prodotti S7 si consiglia di rivolgersi al proprio distributore o al più vicino ufficio vendite Siemens. Poiché i rappresentanti Siemens dispongono di un'adeguata formazione tecnica e di conoscenze specifiche sulle attività, i processi e le esigenze del settore di attività dei clienti, oltre che sui prodotti Siemens, sapranno sicuramente dare una risposta rapida ed efficace a qualsiasi problema.

Indice del contenuto

	Prefazione	3
1	Presentazione del prodotto.....	11
1.1	Introduzione al PLC S7-1200.....	11
1.2	Signal board	13
1.3	Moduli di I/O	14
1.4	Moduli di comunicazione.....	14
1.5	STEP 7 Basic	15
1.5.1	Diversi modi per facilitare il lavoro	15
1.5.2	La Guida sempre a portata di mano	16
1.6	Pannelli di visualizzazione	20
2	Montaggio.....	21
2.1	Procedimenti di installazione e disinstallazione.....	24
2.1.1	Montaggio e smontaggio della CPU	26
2.1.2	Montaggio e smontaggio dei moduli di I/O	28
2.1.3	Montaggio e smontaggio dei moduli di comunicazione	29
2.1.4	Montaggio e smontaggio delle signal board	31
2.1.5	Smontaggio e rimontaggio della morsettiera dell'S7-1200	32
2.2	Istruzioni per il cablaggio	32
3	Concetti base sui PLC	37
3.1	Esecuzione del programma utente	37
3.1.1	Modi di funzionamento della CPU	39
3.1.2	Priorità di esecuzione degli eventi e loro inserimento nella coda d'attesa	43
3.1.3	Memoria della CPU	49
3.1.4	Protezione mediante password della CPU S7-1200.....	54
3.1.5	Ripristino in caso di perdita della password.....	55
3.2	Memorizzazione dei dati, aree di memoria e indirizzamento.....	55
3.3	Tipi di dati.....	60
3.4	Utilizzo della memory card	63
3.4.1	Inserimento di una memory card nella CPU	64
3.4.2	Configurazione del parametro di avvio della CPU prima di copiare il progetto nella memory card	65
3.4.3	Scheda di trasferimento	66
3.4.4	Scheda di programma.....	68

4	Configurazione dei dispositivi	71
4.1	Inserimento di una CPU	72
4.2	Rilevamento della configurazione per una CPU non specificata	73
4.3	Configurazione del funzionamento della CPU	74
4.4	Inserimento di moduli nella configurazione	75
4.5	Configurazione dei parametri dei moduli	76
4.6	Creazione di una connessione di rete	77
4.7	Configurazione di un indirizzo IP del progetto	78
5	Concetti di programmazione	81
5.1	Istruzioni per la progettazione di un sistema PLC	81
5.2	Strutturazione del programma utente	82
5.3	Utilizzo dei blocchi per la strutturazione del programma	83
5.3.1	Blocco organizzativo (OB)	84
5.3.2	Funzione (FC)	86
5.3.3	Blocco funzionale (FB)	86
5.3.4	Blocco dati (DB)	88
5.4	Coerenza dei dati	88
5.5	Selezione del linguaggio di programmazione	89
5.6	Protezione dalla copia	91
5.7	Caricamento degli elementi del programma nella CPU	92
5.8	Caricamento degli elementi del programma dalla CPU	92
5.9	Test del programma	93
6	Istruzioni di programmazione	95
6.1	Istruzioni di base	95
6.1.1	Combinazione logica di bit	95
6.1.1.1	Istruzioni di impostazione e reset	98
6.1.1.2	Istruzioni di fronte di salita e di discesa	100
6.1.2	Temporizzatori	102
6.1.3	Contatori	106
6.1.3.1	Contatori	106
6.1.3.2	Istruzione CTRL_HSC	108
6.1.3.3	Funzionamento del contatore veloce	111
6.1.3.4	Configurazione dell'HSC	113
6.1.4	Confronto	114
6.1.5	Funzioni matematiche	116
6.1.5.1	Istruzione MOD	117
6.1.6	Operazioni di trasferimento	122
6.1.6.1	Istruzione Modifica disposizione	125
6.1.7	Conversione	126
6.1.7.1	Istruzioni Riporta in scala e Normazione	128
6.1.8	Controllo del programma	129
6.1.9	Operazioni logiche	131
6.1.10	Scorrimento e rotazione	134

6.2	Istruzioni avanzate	136
6.2.1	Parametri di errore comuni alle istruzioni avanzate	136
6.2.2	Istruzioni di orologio e calendario	136
6.2.3	Istruzioni con le stringhe e i caratteri	140
6.2.3.1	Descrizione dei dati String	140
6.2.3.2	Istruzioni di conversione di stringhe.....	141
6.2.3.3	Istruzioni con le stringhe	148
6.2.4	Istruzioni di comando del programma.....	154
6.2.4.1	Istruzione Riavvia watchdog del tempo di ciclo	154
6.2.4.2	Istruzione Arresta ciclo di scansione del PLC.....	155
6.2.4.3	Istruzioni di lettura degli errori.....	155
6.2.5	Istruzioni di comunicazione.....	158
6.2.5.1	Comunicazione Ethernet aperta	158
6.2.5.2	Istruzioni punto a punto (PtP)	172
6.2.6	Istruzioni di allarme	172
6.2.6.1	Istruzioni di assegnazione e separazione	172
6.2.6.2	Istruzioni Avvia allarme di ritardo e Annulla allarme di ritardo	176
6.2.6.3	Istruzioni Disabilita allarmi e Abilita allarmi	177
6.2.7	Regolazione PID	178
6.2.8	Istruzioni di controllo del movimento.....	178
6.2.9	Istruzione PWM.....	180
6.2.9.1	Istruzione CTRL_PWM	180
6.3	Istruzioni della biblioteca globale	183
6.3.1	USS.....	183
6.3.1.1	Requisiti per l'utilizzo del protocollo USS.....	183
6.3.1.2	Istruzione USS_DRV.....	186
6.3.1.3	Istruzione USS_PORT	189
6.3.1.4	Istruzione USS_RPM	190
6.3.1.5	Istruzione USS_WPM	191
6.3.1.6	Codici di stato USS	193
6.3.2	MODBUS	194
6.3.2.1	MB_COMM_LOAD	194
6.3.2.2	MB_MASTER.....	196
6.3.2.3	MB_SLAVE	206
7	PROFINET	217
7.1	Comunicazione con un dispositivo di programmazione	219
7.1.1	Realizzazione del collegamento hardware per la comunicazione	219
7.1.2	Configurazione dei dispositivi	220
7.1.3	Assegnazione degli indirizzi IP (Internet Protocol)	220
7.1.3.1	Assegnazione degli indirizzi IP ai dispositivi di programmazione e di rete.....	220
7.1.3.2	Assegnazione dell'indirizzo IP online.....	223
7.1.3.3	Configurazione di un indirizzo IP del progetto	225
7.1.4	Test della rete PROFINET	227
7.2	Comunicazione da HMI a PLC.....	229
7.2.1	Configurazione dei collegamenti logici di rete tra un HMI e una CPU.....	230
7.3	Comunicazione da PLC a PLC	231
7.3.1	Configurazione dei collegamenti logici di rete tra due CPU	232
7.3.2	Configurazione dei parametri di trasmissione (invio) e di ricezione	233
7.3.2.1	Configurazione dei parametri di trasmissione (invio) dell'istruzione TSEND_C	233
7.3.2.2	Configurazione dei parametri di ricezione dell'istruzione TRCV_C	237
7.4	Informazioni di riferimento.....	241
7.4.1	Posizione dell'indirizzo Ethernet (MAC) sulla CPU.....	241
7.4.2	Configurazione della sincronizzazione del Network Time Protocol	242

8	Comunicazioni punto a punto (PtP)	245
8.1	Uso dei moduli di comunicazione RS232 e RS485	245
8.2	Configurazione delle porte di comunicazione	246
8.3	Gestione del controllo di flusso	247
8.4	Configurazione dei parametri di trasmissione (invio) e di ricezione	249
8.5	Programmazione della comunicazione PtP	254
8.5.1	Architettura di interrogazione	255
8.6	Istruzioni punto a punto (PtP)	256
8.6.1	Parametri comuni delle istruzioni punto a punto	256
8.6.2	Istruzione PORT_CFG	258
8.6.3	Istruzione SEND_CFG	260
8.6.4	Istruzione RCV_CFG	261
8.6.5	Istruzione SEND_PTP	268
8.6.6	Istruzione RCV_PTP	271
8.6.7	Istruzione RCV_RST	272
8.6.8	Istruzione SGN_GET	273
8.6.9	Istruzione SGN_SET	274
8.7	Errori	275
9	Tool online e di diagnostica	279
9.1	LED di stato	279
9.2	Collegamento online e connessione a una CPU	281
9.3	Impostazione dell'indirizzo IP e dell'ora	282
9.4	Pannello operatore CPU per la CPU online	282
9.5	Controllo del tempo di ciclo e dell'utilizzo della memoria	283
9.6	Visualizzazione degli eventi di diagnostica nella CPU	283
9.7	Tabelle per il controllo del programma utente	284
A	Dati tecnici	289
A.1	Dati tecnici generali	289
A.2	CPU	294
A.2.1	Dati tecnici della CPU 1211C	294
A.2.2	Dati tecnici della CPU 1212C	300
A.2.3	Dati tecnici della CPU 1214C	305
A.3	Moduli di I/O digitali (SM)	311
A.3.1	Dati tecnici del modulo digitale di ingresso SM 1221	311
A.3.2	Dati tecnici del modulo digitale di uscita SM 1222	313
A.3.3	Dati tecnici del modulo digitale di ingresso/uscita SM 1223	315
A.4	Moduli di I/O analogici (SM)	318
A.4.1	Dati tecnici dei moduli analogici SM 1231, SM 1232, SM 1234	318
A.5	Signal board (SB)	325
A.5.1	Dati tecnici dell'SB 1223 2 ingressi 24 V DC / 2 uscite 24 V DC	325
A.5.2	Dati tecnici dell'SB 1232 1 uscita analogica	327

A.6	Moduli di comunicazione (CM).....	329
A.6.1	CM 1241 RS485, dati tecnici	329
A.6.2	CM 1241 RS232, dati tecnici	330
A.7	Memory card SIMATIC	330
A.8	Simulatori di ingressi.....	331
A.9	Prolunga per I/O.....	332
B	Calcolo del budget di potenza.....	333
B.1	Esempio di calcolo della potenza richiesta	334
B.2	Calcolo della potenza richiesta	335
C	Numeri di ordinazione.....	337
	Indice analitico.....	339

Presentazione del prodotto

1.1 Introduzione al PLC S7-1200

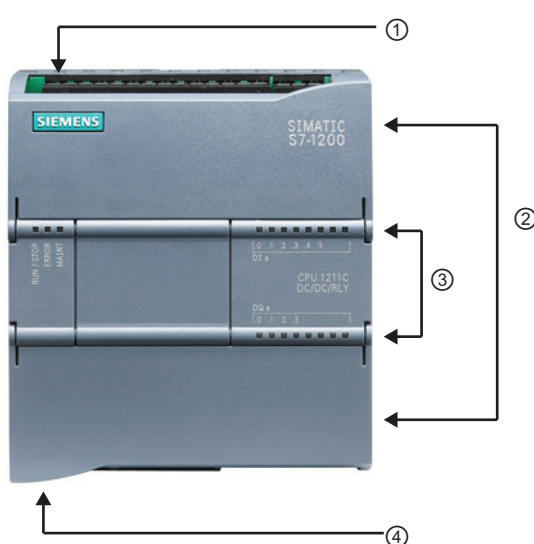
Il controllore a logica programmabile (PLC) S7-1200 è un sistema flessibile e potente in grado di controllare un'ampia varietà di dispositivi e di rispondere alle più diverse esigenze del settore dell'automazione. La struttura compatta, la configurazione flessibile e l'ampio set di operazioni fanno dell'S7-1200 la soluzione ottimale per il controllo di svariate applicazioni.

La CPU riunisce in un'unica apparecchiatura compatta un microprocessore, un alimentatore integrato e circuiti di ingresso e di uscita creando così un potente PLC. Una volta caricato il programma la CPU contiene la logica necessaria per il controllo e il comando dei dispositivi utilizzati nell'applicazione. La CPU controlla gli ingressi e modifica le uscite in base alla logica del programma utente, il quale può comprendere operazioni booleane, di conteggio e di temporizzazione, operazioni matematiche complesse e funzioni per la comunicazione con altri dispositivi intelligenti.

Numerose funzioni di sicurezza contribuiscono a proteggere l'accesso sia alla CPU che al programma di comando:

- Ogni CPU consente di utilizzare una password per configurare l'accesso alle funzioni delle CPU.
- È possibile nascondere il codice in un blocco per proteggere il knowhow dell'autore. Per maggiori informazioni sulla configurazione consultare il paragrafo "Concetti di programmazione (Pagina 97)".

La CPU dispone di una porta PROFINET per la comunicazione tramite rete PROFINET. Sono disponibili moduli per la comunicazione tramite reti RS485 o RS232.



- ① Connettore di alimentazione
- ② Morsettiera estraibile per il cablaggio utente (dietro i coperchi)
- ② Slot per la memory card protetto da un coperchio
- ③ LED di stato per gli I/O on-board
- ④ Connettore PROFINET (in basso nella CPU)

I diversi modelli di CPU sono caratterizzati da una vasta gamma di funzioni e potenzialità, che consentono di realizzare valide soluzioni di automazione per le più diverse applicazioni. Per maggiori informazioni sulle singole CPU consultare i dati tecnici (Pagina 295).

Caratteristica	CPU 1211C	CPU 1212C	CPU 1214C
Dimensioni di ingombro (mm)	90 x 100 x 75		110 x 100 x 75
Memoria utente • Memoria di lavoro • Memoria di caricamento • Memoria a ritenzione	• 25 Kbyte • 1 Mbyte • 2 Kbyte		• 50 Kbyte • 2 Mbyte • 2 Kbyte
I/O on-board locali • Digitali • Analogici	• 6 ingressi/4 uscite • 2 ingressi	• 8 ingressi/6 uscite • 2 ingressi	• 14 ingressi/10 uscite • 2 ingressi
Dimensione dell'immagine di processo	1024 byte di ingressi (I) e 1024 byte di uscite (Q)		
Memoria di merker (M)	4096 byte		8192 byte
Ampliamento con moduli di I/O	Nessuno	2	8
Signal board	1		
Moduli di comunicazione	3 (ampliamento sul lato sinistro)		
Contatori veloci • Monofase • In quadratura di fase	3 • 3 a 100 kHz • 3 a 80 kHz	4 • 3 a 100 kHz 1 a 30 kHz • 3 a 80 kHz 1 a 20 kHz	6 • 3 a 100 kHz 3 a 30 kHz • 3 a 80 kHz 3 a 20 kHz
Uscite di impulsi	2		
Memory card	SIMATIC Memory card (opzionale)		
Tempo di ritenzione dell'orologio hardware	Tipicamente 10 giorni / min. 6 giorni a 40 gradi C.		
PROFINET	1 porta di comunicazione Ethernet		
Velocità di esecuzione operazioni matematiche con numeri reali	18 µs/istruzione		
Velocità di esecuzione operazioni booleane	0,1 µs/istruzione		

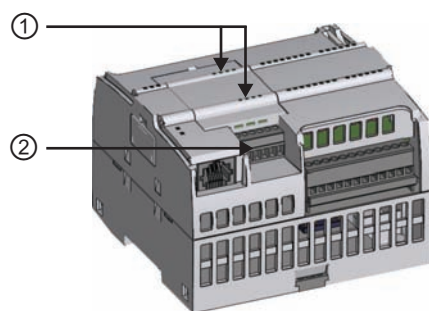
La serie S7-1200 comprende svariati moduli di I/O e signal board che consentono di ampliare le funzionalità della CPU. Sono inoltre disponibili moduli di comunicazioni aggiuntivi che supportano protocolli diversi. Per maggiori informazioni sui singoli moduli consultare i dati tecnici (Pagina 295).

Modulo		Solo ingresso	Solo uscita	Combinazione di I/O
Modulo di I/O (SM)	Digitale	8 ingressi DC	8 uscite DC 8 uscite relè	8 ingressi DC/8 uscite DC 8 ingressi DC/8 uscite relè
		16 ingressi DC	16 uscite DC 16 uscite relè	16 ingressi DC/16 uscite DC 16 ingressi DC/16 uscite relè
	Analogico	4 ingressi analogici/ 8 ingressi analogici	2 uscite analogiche/ 4 uscite analogiche	4 ingressi analogici/2 uscite analogiche
Signal board (SB)	Digitale	-	-	2 ingressi DC/2 uscite DC
	Analogico	-	1 uscita analogica	-
Modulo di comunicazione (CM)				
• RS485 • RS232				

1.2 Signal board

Le signal board (SB) consentono di aggiungere ingressi e uscite alla CPU. Le SB possono disporre di I/O digitali o analogici e vengono installate sul lato anteriore della CPU.

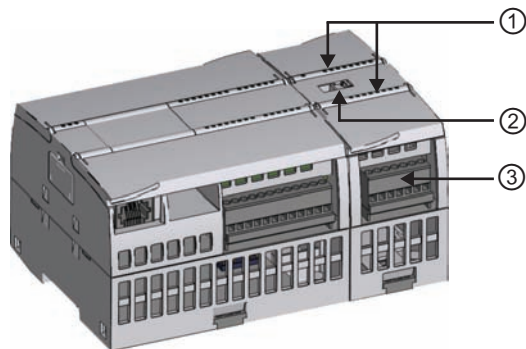
- SB con 4 I/O digitali (2 ingressi DC e 2 uscite DC)
- SB con 1 uscita analogica



- ① LED di stato dell'SB
- ② Morsettiera estraibile per il cablaggio utente

1.3 Moduli di I/O

I moduli di I/O consentono di ampliare la funzionalità della CPU e vanno collegati a destra della CPU.

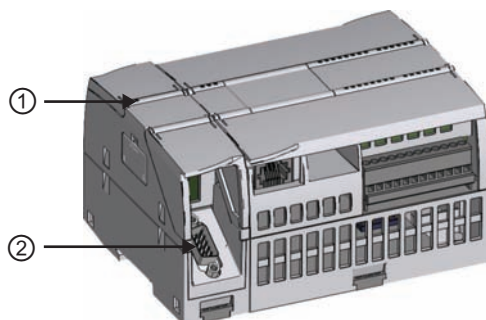


- ① LED di stato degli I/O del modulo
- ② Connettore di bus
- ③ Morsettiera estraibile per il cablaggio utente

1.4 Moduli di comunicazione

La serie S7-1200 comprende moduli di comunicazione (CM) che consentono di ampliare la funzionalità del sistema. Sono disponibili due tipi di moduli: RS232 e RS485.

- La CPU supporta fino a 3 moduli di comunicazione
- che vengono collegati a sinistra della CPU (o di un altro modulo CM).



- ① LED di stato del modulo di comunicazione
- ② Connettore di comunicazione

1.5 STEP 7 Basic

Il software STEP 7 Basic mette a disposizione un ambiente intuitivo in cui sviluppare, modificare e controllare la logica per il comando della propria applicazione, compresi gli strumenti necessari per gestire e configurare i dispositivi del progetto, quali i PLC e gli HMI. STEP 7 Basic consente di utilizzare due linguaggi di programmazione (KOP e FUP) per sviluppare in modo efficiente e pratico il programma di comando per la propria applicazione e fornisce gli strumenti per creare e progettare i dispositivi HMI del progetto.

STEP 7 Basic mette a disposizione una Guida in linea molto ampia che facilita la ricerca delle informazioni.

Per installare STEP 7 Basic inserire il CD nel drive per CD ROM del computer. Il programma di installazione si avvia automaticamente e guida l'utente fino al termine della procedura. Per maggiori informazioni sull'argomento consultare il file Leggimi.

Nota

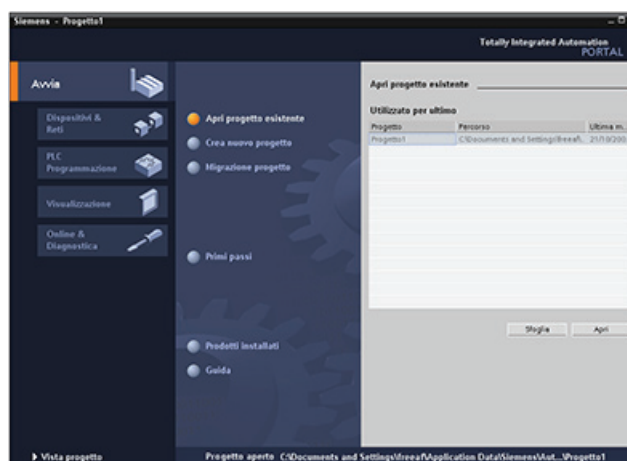
Per poter installare il software STEP 7 Basic in un PC con sistema operativo Windows 2000, Windows XP o Windows Vista è necessario connettersi con i privilegi di amministratore.

1.5.1 Diversi modi per facilitare il lavoro

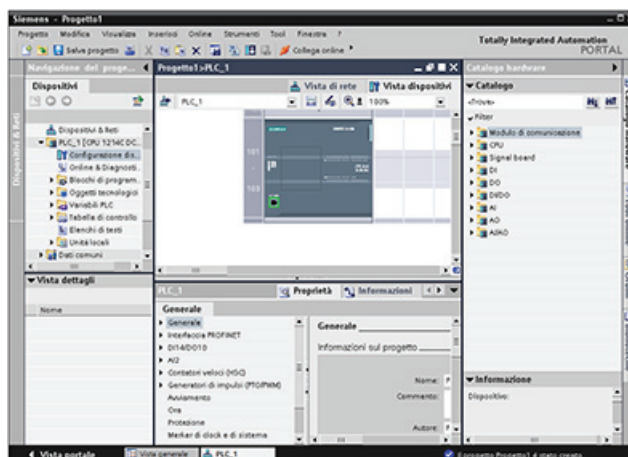
Totally Integrated Automation Portal mette a disposizione due diverse viste del set di strumenti che consentono di lavorare in modo più efficiente: un set di portali orientati al task e organizzati in base alla funzione degli strumenti (vista portale) e una vista degli elementi orientata al progetto (vista progetto). L'utente ha la possibilità di scegliere quale delle due viste utilizzare per operare in modo più efficiente. Per passare da una vista all'altra basta un clic con il mouse.

La vista portale mette a disposizione una rappresentazione funzionale dei task del progetto e organizza le funzioni degli strumenti in base ai task da eseguire, ad es. creare la configurazione dei componenti hardware e delle reti.

Capire come procedere e quali task scegliere è estremamente facile.



La vista progetto consente di accedere ai componenti di un progetto. La possibilità di visualizzare tutti i componenti insieme consente di accedere facilmente a tutti gli aspetti del progetto. Il progetto contiene tutti gli elementi creati o conclusi.



1.5.2 La Guida sempre a portata di mano

Una risposta rapida in caso di dubbi

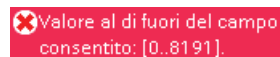
STEP 7 Basic dispone di una Guida contestuale intelligente che consente di risolvere gli eventuali problemi in modo rapido ed efficiente:

- Alcuni campi di immissione dispongono di una casella a discesa (roll-out) che facilita l'inserimento delle informazioni corrette (campi di valori e tipi di dati validi). Se, ad esempio, si indica un valore errato si apre una casella di testo che visualizza il campo di valori validi.
- Alcune descrizioni comando dell'interfaccia (ad es. quelle delle istruzioni) si "aprono" e visualizzano informazioni aggiuntive. A volte le caselle a discesa contengono link ad argomenti specifici del sistema di informazione online (Guida in linea).

Inoltre STEP 7 Basic dispone di un sistema di informazione molto ampio che descrive la funzionalità degli strumenti SIMATIC.

Roll-out e descrizioni comando con casella a discesa

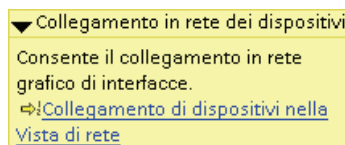
I campi di alcune finestre di dialogo e task card reagiscono all'input dell'utente visualizzando una casella di testo che indica il campo di valori validi o i tipi di dati richiesti.



Valore al di fuori del campo consentito: [0..8191]

Gli elementi dell'interfaccia software visualizzano descrizioni comando per spiegare la propria funzione. Alcuni di essi, quali le icone "Apri" e "Salva" non contengono ulteriori informazioni. Altri elementi dispongono invece di un sistema che consente di visualizzare una descrizione più ampia in una casella che si "apre" sotto la descrizione comando (un triangolo nero in alto nella descrizione comando indica che sono disponibili informazioni aggiuntive).

La descrizione comando di un elemento dell'interfaccia software compare quando ci si posiziona il cursore. Per visualizzare ulteriori informazioni basta puntare il cursore sulla descrizione comando. A volte le caselle a discesa contengono link ad argomenti specifici del sistema di informazione. Facendo clic sul link si richiama l'argomento specifico.



▼ Collegamento in rete dei dispositivi

Consente il collegamento in rete grafico di interfacce.

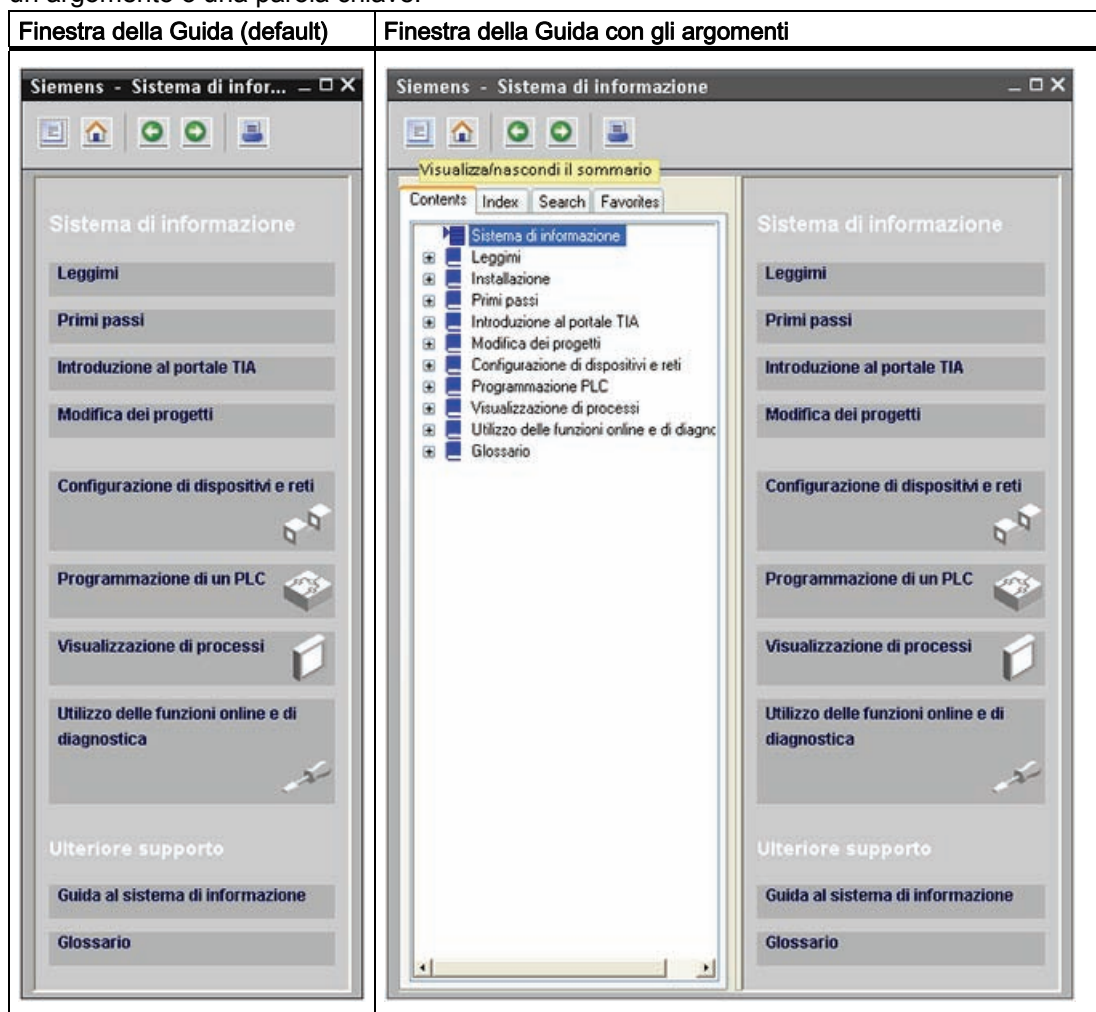
⇒ [Collegamento di dispositivi nella Vista di rete](#)

Sistema di informazione

STEP 7 Basic mette a disposizione informazioni online molto ampie e un sistema di Guida che descrive tutti i prodotti SIMATIC installati. Il sistema comprende anche informazioni di riferimento ed esempi. Per accedervi procedere in uno dei seguenti modi:

- Nella vista portale: selezionare il portale Start e fare clic sul comando "?".
- Nella vista progetto: selezionare il comando "Visualizza la Guida" nel menu "?".
- Nelle descrizioni comando con casella a discesa: fare clic su un link per visualizzare ulteriori informazioni sull'argomento.

Il sistema di informazione si apre in una finestra che non si sovrappone alle aree di lavoro. Fare clic sul pulsante "Visualizza/nascondi il sommario" del sistema di informazione per visualizzarne gli argomenti e disancorare la finestra della Guida. È quindi possibile modificare le dimensioni della finestra della Guida. Le schede "Argomenti della Guida" o "Sommario" consentono di effettuare una ricerca nel sistema di informazione specificando un argomento o una parola chiave.

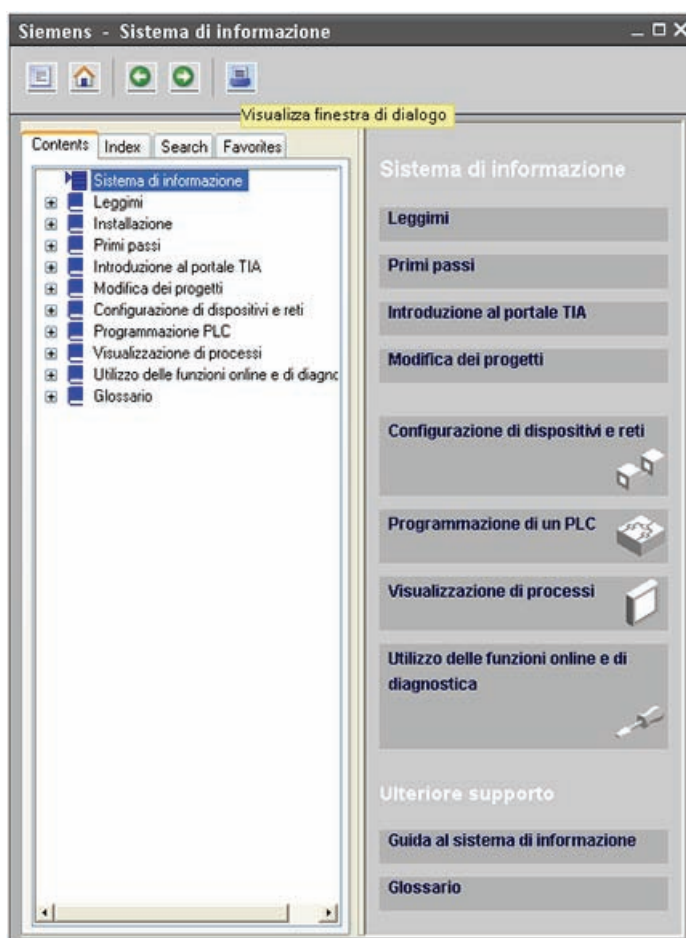


Nota

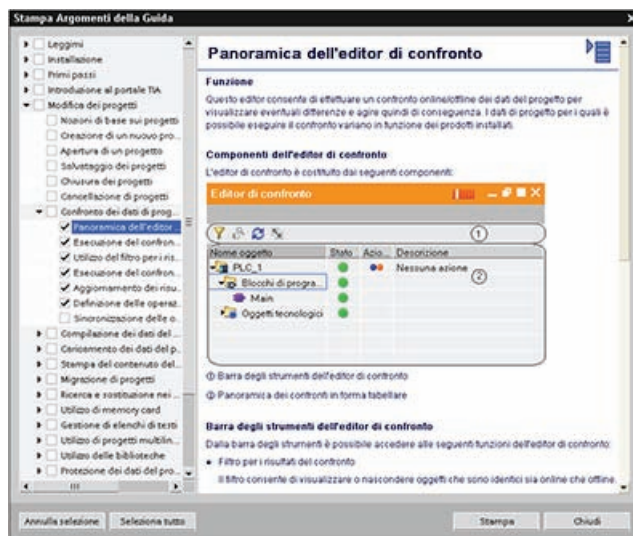
Se STEP 7 Basic è ingrandito al massimo, il pulsante "Visualizza/nascondi il sommario" non consente di disancorare la finestra della Guida. Per disancorarla si dovrà cliccare il pulsante "Ingrandisci" e poi spostarla e modificarne le dimensioni.

Stampa degli argomenti del sistema di informazione

Per stampare il contenuto del sistema di informazione cliccare sul pulsante "Stampa" della finestra della Guida.



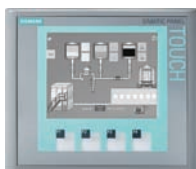
Per stampare il contenuto del sistema di informazione cliccare sul pulsante "Stampa" della finestra della Guida.



La finestra di dialogo "Stampa" consente di selezionare gli argomenti da stampare. Accertarsi che il pannello visualizzi un argomento. Eventualmente selezionare un qualsiasi altro argomento. Cliccare sul pulsante "Stampa" per inviare alla stampante gli argomenti selezionati.

1.6 Pannelli di visualizzazione

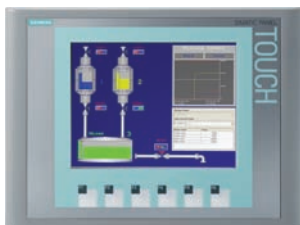
Considerato che la visualizzazione sta diventando un elemento standard nella maggior parte delle macchine, i SIMATIC HMI Basic Panel dispongono di schermi a sfioramento per le principali operazioni di comando e di controllo dell'operatore. Tutti i pannelli presentano un grado di protezione IP65 e sono dotati di certificazione CE, UL, cULus e NEMA 4x.



KTP 400 Basic PN

- Mono (STN, scala di grigi)
- Schermo a sfioramento 4" + 4 tasti con feedback tattile
- Ritratto o paesaggio
- Dimensione: 3.8"
- Risoluzione: 320 x 240

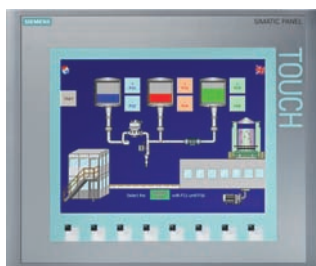
- 128 variabili
- 50 pagine di processo
- 200 allarmi
- 25 curve
- 32 KB di memoria ricette
- 5 ricette, 20 record di dati, 20 voci



KTP 600 Basic PN

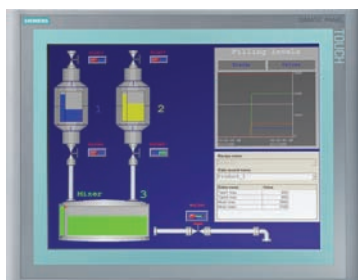
- Colore (TFT, 256 colori) o mono (STN, scala di grigi)
- Schermo a sfioramento 6" + 6 tasti con feedback tattile
- Ritratto o paesaggio
- Dimensione: 5.7"
- Risoluzione: 320 x 240

- 128 variabili
- 50 pagine di processo
- 200 allarmi
- 25 curve
- 32 KB di memoria per le ricette
- 5 ricette, 20 record di dati, 20 voci



KTP1000 Basic PN

- Colore (TFT, 256 colori)
- Schermo a sfioramento 10" + 8 tasti con feedback tattile
- Dimensione: 10.4"
- Risoluzione: 640 x 480
- 256 variabili
- 50 pagine di processo
- 200 allarmi
- 25 curve
- 32 KB di memoria per le ricette
- 5 ricette, 20 record di dati, 20 voci



TP1500 Basic PN

- Colore (TFT, 256 colori)
- Schermo a sfioramento 15"
- Dimensione: 15.1"
- Risoluzione: 1024 x 768
- 256 variabili
- 50 pagine di processo
- 200 allarmi
- 25 curve
- 32 KB di memoria ricette (flash integrata)
- 5 ricette, 20 record di dati, 20 voci

Montaggio

I dispositivi S7-1200 sono estremamente semplici da installare. Possono essere montati su un pannello o una guida DIN standard e orientati sia in senso orizzontale che verticale. Le ridotte dimensioni dell'S7-1200 permettono inoltre un uso più razionale dello spazio.

AVVERTENZA

I PLC SIMATIC S7-1200 sono controllori di tipo aperto. Devono essere installati in una custodia, un armadio o una sala di controllo il cui accesso sia consentito esclusivamente al personale autorizzato.

Il mancato rispetto di questi requisiti di installazione può causare la morte o gravi lesioni alle persone e/o danni alle cose.

In fase di installazione dei PLC S7-1200 rispettare sempre i requisiti qui descritti.

Isolare i dispositivi S7-1200 dal calore, dall'alta tensione e dal rumore elettrico.

Una regola generale a cui attenersi durante il montaggio è quella di separare i dispositivi che generano alta tensione e un elevato rumore elettrico dai dispositivi logici che funzionano con basse tensioni, quali l'S7-1200.

Quando si configura la disposizione dell'S7-1200 nel pannello è bene individuare i dispositivi che emettono calore e dislocare quelli elettronici nelle zone meno calde dell'armadio. Riducendo l'esposizione alle alte temperature si garantisce una maggiore durata dei dispositivi elettronici.

Va inoltre considerata la disposizione dei conduttori dei dispositivi nel pannello. È importante non disporre i conduttori di segnale a bassa tensione e i cavi di comunicazione assieme ai conduttori di potenza AC e ai conduttori DC ad alta corrente e a commutazione rapida.

Prevedere uno spazio libero adeguato per il raffreddamento e il cablaggio

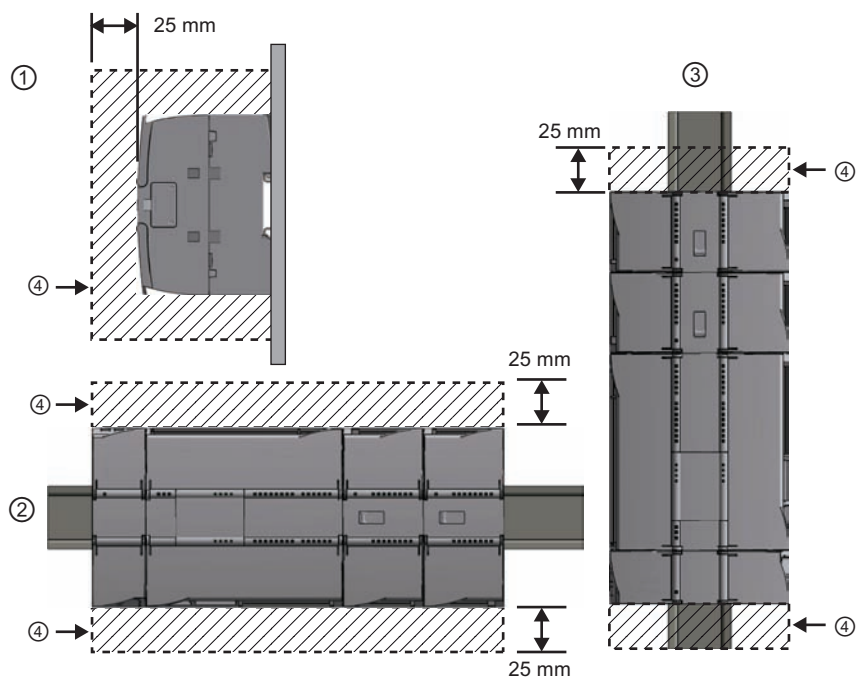
I dispositivi S7-1200 sono stati progettati per il raffreddamento a convezione naturale. Per garantire un raffreddamento corretto è necessario lasciare uno spazio libero di almeno 25 mm sia sopra che sotto i dispositivi. Lasciare inoltre almeno 25 mm di spazio libero tra il lato anteriore dei moduli e l'interno della custodia.



CAUTELA

In caso di montaggio verticale, la temperatura ambiente massima va ridotta di 10 °C. Il sistema S7-1200 deve essere orientato in modo che la CPU sia l'ultima in basso.

Quando si progetta la disposizione del sistema S7-1200 si deve prevedere uno spazio libero sufficiente per il cablaggio e il collegamento dei cavi di comunicazione.



① Vista laterale

② Montaggio orizzontale

③ Montaggio verticale

④ Spazio libero

Budget di potenza

La CPU dispone di un alimentatore integrato che provvede ad alimentare la CPU, i moduli di I/O, le signal board, gli altri moduli di comunicazione e le altre utenze che richiedono un'alimentazione a 24 V DC.

Per maggiori informazioni sul budget di potenza di 5 V DC fornito dalla CPU e richiesto dai moduli di I/O, dalle signal board e dai moduli di comunicazione consultare i dati tecnici (Pagina 295). Per determinare quanta potenza (o corrente) la CPU è in grado di erogare alla configurazione progettata consultare "Calcolo del budget di potenza" (Pagina 339).

La CPU dispone inoltre di un'alimentazione per sensori di 24 V DC che fornisce una tensione continua a 24 V DC agli ingressi, alla bobina a relè di potenza dei moduli di I/O o ad altri dispositivi. Se la potenza di 24 V DC richiesta è superiore a quella fornita dall'alimentazione per sensori è necessario aggiungere al sistema un'unità di alimentazione esterna da 24 V DC. Per maggiori informazioni sul budget di potenza per sensori a 24 V DC della CPU S7-1200 utilizzare i dati tecnici (Pagina 295).

Se si utilizza un'unità di alimentazione esterna da 24 V DC, accertarsi che non sia collegata in parallelo all'alimentazione per sensori della CPU. Per migliorare la protezione dal rumore elettrico si consiglia di collegare il polo comune (M) dei diversi alimentatori.

AVVERTENZA

Se si collega un alimentatore esterno a 24 V DC in parallelo all'alimentazione per sensori a 24 V DC, può verificarsi un conflitto tra le due alimentazioni che cercheranno di imporre il proprio livello di tensione di uscita preferenziale.

Ne potrebbero derivare una riduzione della durata o il guasto immediato di uno o entrambi gli alimentatori, con conseguente funzionamento imprevedibile del sistema. Ciò può causare la morte o gravi lesioni alle persone e/o danni alle cose.

L'alimentazione per sensori DC e le unità di alimentazione esterna devono alimentare I/O diversi.

Alcune porte di ingresso dell'alimentazione a 24 V DC del sistema S7-1200 sono interconnesse, ovvero un circuito logico comune collega tra loro più morsetti M. Sono interconnessi, ad esempio, i seguenti circuiti, se contrassegnati come "non isolati" nelle schede tecniche: l'alimentazione a 24 V DC della CPU, l'ingresso di alimentazione della bobina del relè di un SM o di un ingresso non isolato. Tutti i morsetti M non isolati devono essere collegati allo stesso potenziale di riferimento esterno.

AVVERTENZA

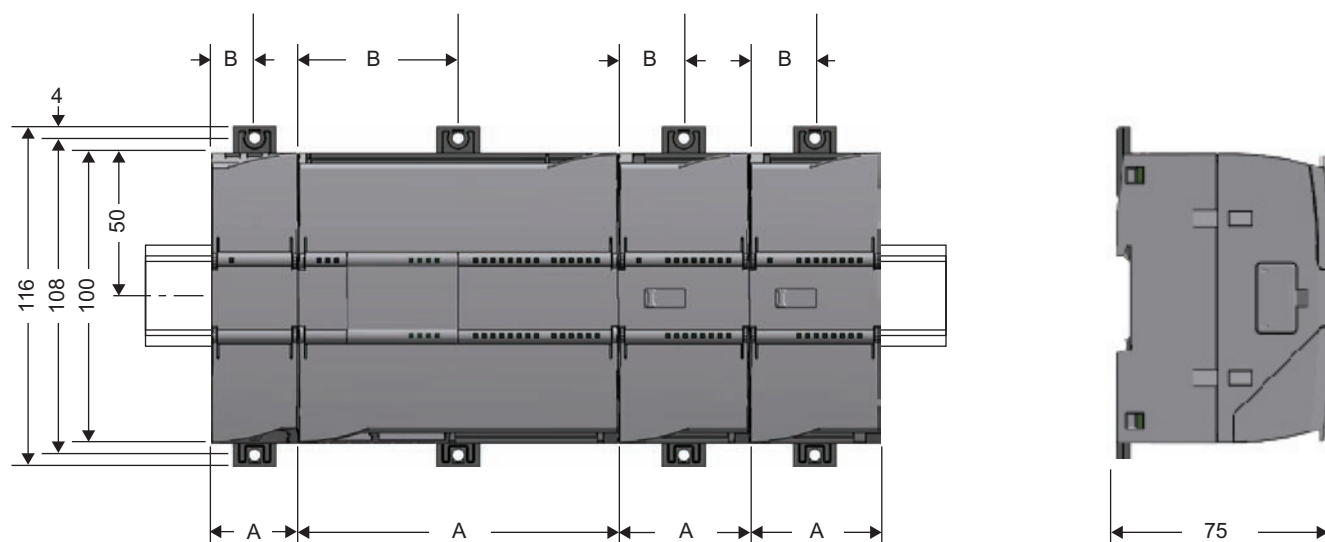
Se si collegano i morsetti M non isolati a potenziali di riferimento diversi si formano flussi di corrente indesiderati che possono danneggiare il PLC e le apparecchiature a cui è collegato o farli funzionare in modo imprevedibile.

Il mancato rispetto di queste istruzioni può provocare danni o un funzionamento imprevisto e causare la morte o gravi lesioni alle persone e/o danni alle cose.

È quindi importante accertarsi che i morsetti M non isolati del sistema S7-1200 siano collegati allo stesso potenziale di riferimento.

2.1 Procedimenti di installazione e disinstallazione

Quote di montaggio (mm)



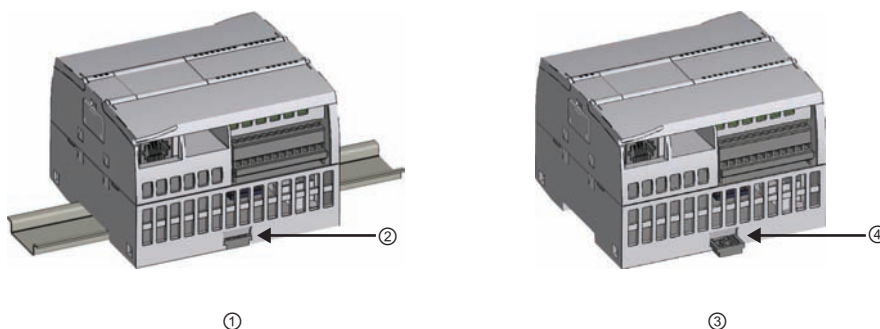
Dispositivi S7-1200		Larghezza A	Larghezza B
CPU:	CPU 1211C e CPU 1212C	90 mm	45 mm
	CPU 1214C	110 mm	55 mm
Moduli di I/O:	8 e 16 I/O DC e relè (8I, 16I, 8Q, 16Q, 8I/8Q)	45 mm	22,5 mm
	Analogici (4AI, 8AI, 4AI/4AQ, 2AQ, 4AQ)		
	16I/16Q relè (16I/16Q)	70 mm	35 mm
Moduli di comunicazione:	CM 1241 RS232 e CM 1241 RS485	30 mm	15 mm

Le CPU, gli SM e i CM possono essere montati sia su guida DIN che su pannello. Per fissare il dispositivo alla guida DIN si utilizzano gli appositi ganci. Questi possono essere anche estratti e impiegati come punti di fissaggio delle viti per montare l'unità direttamente sul pannello. Il foro per i ganci DIN sul dispositivo ha una dimensione interna di 4,3 mm.

È necessario prevedere una zona termica di 25 mm sopra e sotto l'unità per consentire il passaggio dell'aria.

Montaggio e smontaggio dei dispositivi S7-1200

La CPU può essere montata in modo estremamente semplice su una guida DIN standard o un pannello. Il dispositivo dispone di ganci per il fissaggio alla guida DIN che possono essere anche estratti e impiegati come punti di fissaggio delle viti per montare l'unità sul pannello.



① Montaggio su guida DIN

② Gancio per guida DIN in posizione di bloccaggio

③ Montaggio su pannello

④ Gancio estratto per il montaggio su pannello

Prima di installare o disinstallare dei dispositivi elettrici accertarsi che siano spenti. Controllare inoltre che sia stata disinserita l'alimentazione dalle eventuali apparecchiature collegate.

AVVERTENZA

Il montaggio o lo smontaggio dell'S7-1200 e delle relative apparecchiature in presenza di alimentazione può provocare scosse elettriche o il funzionamento imprevisto delle apparecchiature.

La mancata disinserizione dell'alimentazione dall'S7-1200 e da tutte le apparecchiature collegate durante il montaggio o lo smontaggio può provocare la morte o gravi lesioni alle persone e/o danni alle cose a causa di scosse elettriche o del funzionamento imprevisto delle apparecchiature.

Attenersi sempre alle norme di sicurezza e accertarsi che l'S7-1200 sia isolata dall'alimentazione prima di installare o disinstallare le CPU S7-1200 o le apparecchiature collegate.

Quando si sostituisce o si monta un S7-1200 accertarsi di aver scelto il tipo di modulo corretto o un dispositivo equivalente.



AVVERTENZA

Il montaggio errato di un'unità S7-1200 può determinare un funzionamento anomalo del programma dell'S7-1200.

La sostituzione di un dispositivo S7-1200 con un modello diverso o il suo errato posizionamento possono causare la morte, gravi lesioni personali e/o danni alle apparecchiature a causa del funzionamento imprevisto delle stesse.

Sostituire l'S7-1200 con un dispositivo dello stesso modello e accertarsi di averlo collocato nella posizione corretta.

2.1.1 Montaggio e smontaggio della CPU

Montaggio

La CPU può essere montata su un pannello o una guida DIN.

Nota

Collegare i moduli di comunicazione alla CPU e montare l'intero gruppo di elementi come si trattasse di un'unica unità. Montare i moduli di I/O separatamente, dopo aver installato la CPU.

Per montare la CPU su un pannello procedere nel seguente modo:

1. Praticare e filettare i fori di montaggio (M4 o American Standard numero 8) rispettando le quote indicate.
2. Estrarre dal modulo i ganci di montaggio. Accertarsi di aver estratto tutti i ganci per la guida DIN, sia quelli sopra che quelli sotto la CPU.
3. Fissare l'unità al pannello di montaggio inserendo le viti nei ganci.

Nota

Se si monta il sistema in un ambiente soggetto a forti vibrazioni o in senso verticale, il montaggio dell'S7-1200 su pannello garantisce una maggiore protezione.

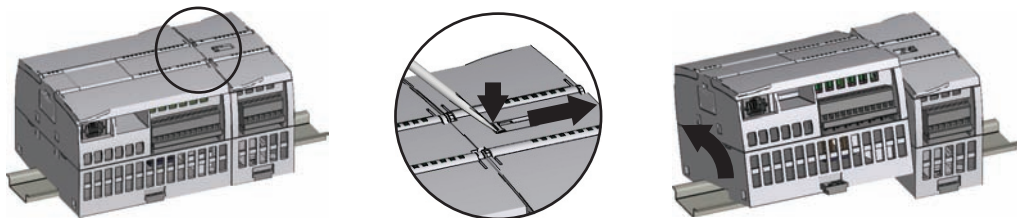
Per montare la CPU su una guida DIN procedere nel seguente modo:



1. Montare la guida DIN. Fissare la guida al pannello di montaggio ogni 75 mm.
2. Agganciare la CPU sopra la guida DIN.
3. Estrarre il gancio per la guida DIN sotto la CPU in modo che questa possa adattarsi alla guida.
4. Ruotare la CPU verso il basso per posizionarla sulla guida.
5. Premere i ganci in modo da bloccare la CPU alla guida.

Smontaggio

Predisporre la CPU per lo smontaggio disinserendo l'alimentazione dalla CPU e scollegando i morsetti di I/O, i conduttori e i cavi. Smontare la CPU e gli eventuali moduli di comunicazione collegati come fossero un'unica unità. Tutti i moduli di I/O devono restare montati.



Se è stato collegato un modulo di I/O alla CPU, disinserire il connettore di bus:

1. Inserire un cacciavite a lato della linguetta posta sopra il modulo di I/O.
2. Premere verso il basso per sganciare il connettore dalla CPU.
3. Far scorrere la linguetta completamente verso destra.

Smontare la CPU:

1. Estrarre il gancio per la guida DIN in modo da sbloccare la CPU dalla guida.
2. Ruotare la CPU verso l'alto e sganciarla dalla guida, quindi smontarla dal sistema.

2.1.2 Montaggio e smontaggio dei moduli di I/O

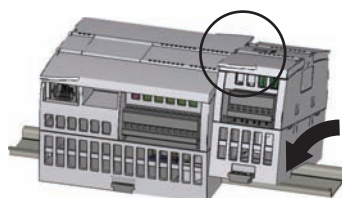
Montaggio

Gli SM possono essere montati dopo aver installato la CPU.



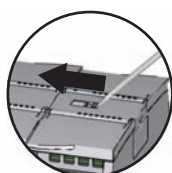
Togliere il coperchio del connettore sul lato destro della CPU.

- Inserire un cacciavite nell'intaglio sopra il coperchio.
- Far leva con delicatezza sul coperchio per toglierlo. Conservarlo per poterlo eventualmente riutilizzare.



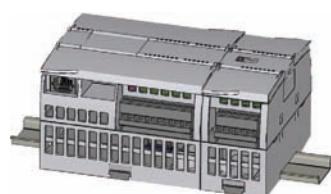
Collocare l'SM a fianco della CPU.

1. Agganciare l'SM sopra la guida DIN.
2. Estrarre il gancio per la guida DIN posto in basso in modo che l'SM possa incastrarsi sulla guida.
3. Ruotare l'SM verso il basso in modo da posizionarla a fianco della CPU e premere il gancio in alto per bloccarla sulla guida.



Inserire il connettore di bus.

1. Inserire un cacciavite a lato della linguetta posta sopra l'SM.
2. Spingere completamente la linguetta verso sinistra per inserire il connettore di bus nella CPU.



Con questa operazione si realizzano i collegamenti sia meccanici che elettrici per l'SM.

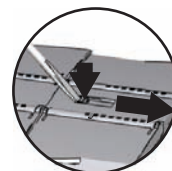
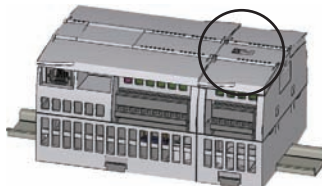
La stessa procedura consente di installare un modulo di I/O in un modulo I/O.

Smontaggio

Gli SM possono essere smontati senza dover smontare anche la CPU o altri eventuali SM. Predisporre l'SM per lo smontaggio disinserendo l'alimentazione dalla CPU e scollegando i morsetti di I/O e i conduttori dall'SM.

Disinserire il connettore di bus.

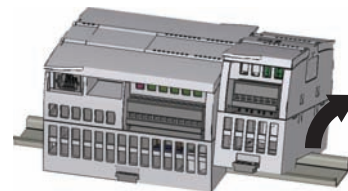
1. Inserire un cacciavite a lato della linguetta posta sopra l'SM.
2. Premere verso il basso per sganciare il connettore dalla CPU.
3. Far scorrere la linguetta completamente verso destra.



Se è presente un altro SM sulla destra ripetere le operazioni anche per questo modulo.

Smontare l'SM:

1. Estrarre il gancio per la guida DIN posto in basso in modo da sganciare l'SM dalla guida.
2. Ruotare l'SM verso l'alto e toglierla dalla guida. Smontare l'SM dal sistema.
3. Se necessario rimettere il coperchio sul connettore di bus della CPU per evitare che vi entri della sporcizia.



La stessa procedura consente di disinstallare un modulo di I/O da un modulo I/O.

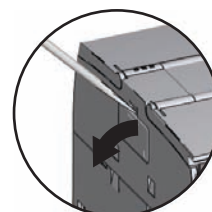
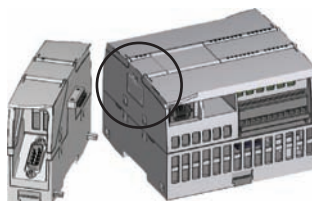
2.1.3 Montaggio e smontaggio dei moduli di comunicazione

Montaggio

Innanzitutto è necessario collegare il CM alla CPU, quindi si può montare l'intero gruppo di elementi sulla guida DIN o sul pannello come si trattasse di un'unica unità.

Togliere il coperchio del bus dal lato sinistro della CPU:

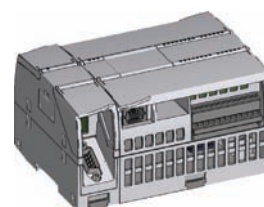
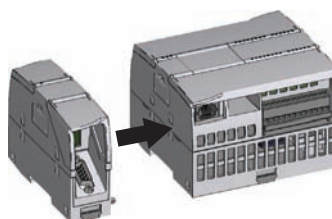
1. Inserire un cacciavite nell'incasso sopra il coperchio del bus.
2. Far leva con delicatezza sul bordo superiore del coperchio.



Togliere il coperchio del bus. Conservarlo per poterlo eventualmente riutilizzare.

Collegare le unità:

1. Allineare il connettore di bus e i perni del CM con i fori della CPU
2. Premere le unità una contro l'altra finché i perni scattano in posizione.



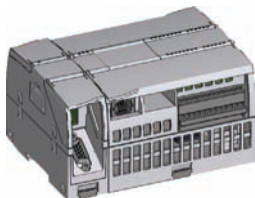
Montaggio delle unità sulla guida DIN o su pannello.

1. Per il montaggio sulla guida DIN accertarsi che, sia nella CPU che nei CM collegati, il gancio superiore sia bloccato (in posizione rientrata) e quello inferiore estratto.
2. Montare la CPU e i CM collegati come indicato in Montaggio e smontaggio della CPU (Pagina 29).
3. Una volta montati i dispositivi sulla guida DIN fissarli spingendo i ganci inferiori in posizione di bloccaggio.

Per il montaggio su pannello verificare che i ganci siano estratti.

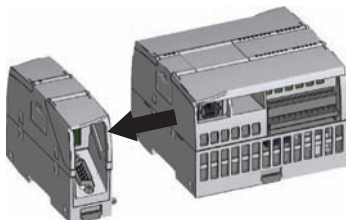
Smontaggio

La CPU e i CM possono essere smontati dalla guida DIN o dal pannello come fossero un'unica unità.



Prima di smontare il CM eseguire le seguenti operazioni preliminari.

1. Scollegare la CPU dall'alimentazione.
2. Scollegare i morsetti di I/O, tutti i conduttori e i cavi dalla CPU e dai CM.
3. Per il montaggio sulla guida DIN estrarre i ganci inferiori della CPU e dei CM.
4. Smontare la CPU e i CM dalla guida DIN o dal pannello.



Smontare il CM.

1. Afferrare la CPU e i CM tenendoli ben saldi.
2. Staccarli.

Non utilizzare attrezzi per separare i moduli perché potrebbero subire dei danni.

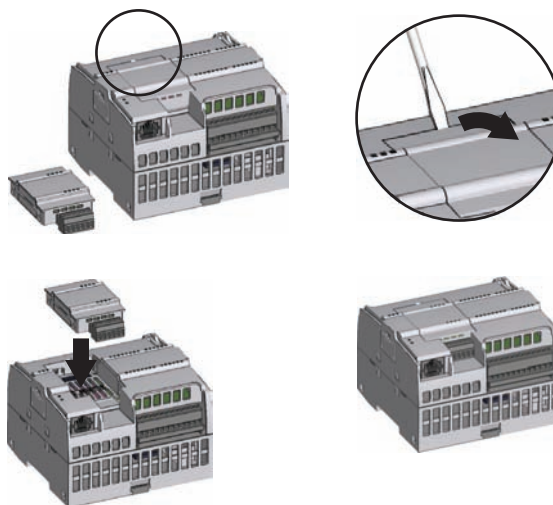
2.1.4 Montaggio e smontaggio delle signal board

Montaggio

Predisporre la CPU per l'installazione dell'SB disinserendo l'alimentazione dalla CPU e togliendo i coperchi della morsettiera superiore e inferiore della CPU.

Per montare l'SB procedere nel seguente modo:

1. Inserire un cacciavite nell'intaglio dietro il coperchio posto sopra la CPU.
2. Far leva con delicatezza sul coperchio e toglierlo dalla CPU.
3. Inserire l'SB nella sua sede di montaggio sopra la CPU.
4. Premere con forza l'SB finché non scatta in posizione.
5. Rimontare i coperchi delle morsettiere.

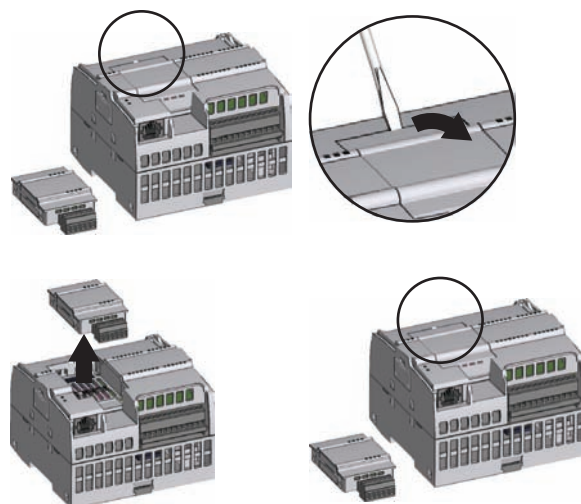


Smontaggio

Predisporre la CPU per lo smontaggio dell'SB disinserendo l'alimentazione dalla CPU e togliendo i coperchi della morsettiera superiore e inferiore della CPU.

Per smontare l'SB procedere nel seguente modo:

1. Inserire un cacciavite nell'intaglio posto sopra l'SB.
2. Far leva con delicatezza sull'SB per sganciarla dalla CPU.
3. Estrarre l'SB dalla sede di montaggio sopra la CPU.
4. Rimontare il coperchio dell'SB.
5. Rimontare i coperchi delle morsettiere.



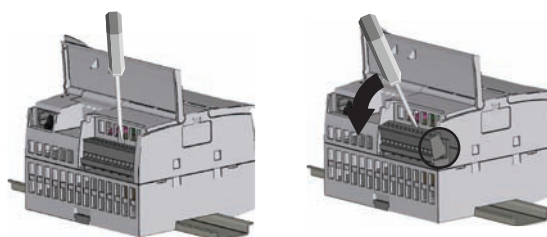
2.1.5 Smontaggio e rimontaggio della morsettiera dell'S7-1200

La CPU, gli SB e gli SM dispongono di una morsettiera estraibile che facilita il collegamento dei conduttori. Prima di smontare la morsettiera è necessario predisporre il sistema nel seguente modo:

- Scollegare la CPU dall'alimentazione.
- Aprire il coperchio della morsettiera.

Per smontare la morsettiera procedere nel seguente modo:

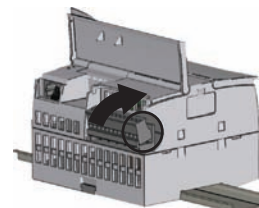
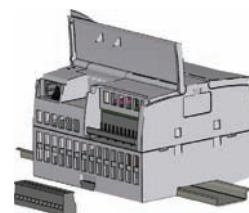
1. Ispezionare il bordo superiore della morsettiera e cercare l'intaglio in cui inserire il cacciavite.
2. Inserire la lama del cacciavite nell'intaglio.
3. Far leva con delicatezza sul bordo superiore della morsettiera spingendola fuori dalla CPU. La morsettiera si sgancia con uno scatto.
4. Afferrare la morsettiera ed estrarla dalla CPU.



Per montare la morsettiera procedere nel seguente modo:

1. Preparare i componenti per il montaggio della morsettiera disinserendo l'alimentazione dalla CPU e aprendo il coperchio della morsettiera.
2. Allineare la morsettiera ai perni dell'unità.
3. Allineare il lato di cablaggio con il bordo della base della morsettiera.
4. Premere con forza e ruotare la morsettiera fino ad agganciarla in posizione.

Verificare che sia allineata e innestata correttamente.



2.2 Istruzioni per il cablaggio

Una messa a terra e un cablaggio corretti sono indispensabili per garantire il funzionamento ottimale del sistema e proteggere adeguatamente l'applicazione e l'S7-1200 dal rumore elettrico. Consultare gli schemi elettrici dell'S7-1200 riportati nei dati tecnici (Pagina 295).

Presupposti

Prima di mettere a terra o cablare dei dispositivi elettrici accertarsi che siano spenti. Controllare inoltre che sia stata disinserita l'alimentazione dalle eventuali apparecchiature collegate.

Per il cablaggio dell'S7-1200 e delle apparecchiature collegate attenersi alle normative vigenti. L'installazione e l'utilizzo del sistema devono essere conformi alle norme nazionali e locali. Rivolgersi alle autorità locali competenti per stabilire quali norme e regolamenti siano applicabili al caso specifico.

AVVERTENZA

Il montaggio o il cablaggio dell'S7-1200 e delle relative apparecchiature in presenza di alimentazione può provocare scosse elettriche o il funzionamento imprevisto delle apparecchiature. La mancata disinserizione dell'alimentazione dall'S7-1200 e da tutte le apparecchiature collegate durante l'installazione o la disinstallazione può provocare la morte o gravi lesioni alle persone e/o danni alle cose a causa di scosse elettriche o del funzionamento imprevisto delle apparecchiature.

Attenersi sempre alle norme di sicurezza e accertarsi che l'S7-1200 sia isolata dall'alimentazione prima di installare o disinstallare l'S7-1200 o le apparecchiature collegate.

Tener sempre in considerazione la sicurezza quando si effettua la messa a terra e il cablaggio dell'S7-1200. I dispositivi elettronici di controllo come l'S7-1200 possono guastarsi e causare il funzionamento imprevisto delle apparecchiature comandate o controllate. È quindi indispensabile prevedere delle protezioni che siano indipendenti dall'S7-1200 per evitare possibili lesioni alle persone e danni alle apparecchiature.

AVVERTENZA

In condizioni non sicure i dispositivi di comando possono funzionare in modo errato e determinare un funzionamento scorretto delle apparecchiature comandate. Ciò può causare la morte o gravi lesioni alle persone e/o danni alle cose.

Prevedere una funzione di arresto d'emergenza, dispositivi elettromeccanici di esclusione o altre protezioni ridondanti che siano indipendenti dall'S7-1200.

Istruzioni per l'isolamento

I separatori per l'isolamento dell'alimentatore AC dell'S7-1200 e per l'isolamento degli I/O verso i circuiti AC sono stati progettati e omologati per garantire un isolamento sicuro tra le tensioni della linea AC e i circuiti a bassa tensione. Questi separatori comprendono un isolamento doppio o rinforzato oppure un isolamento di base associato ad uno supplementare, a seconda della norma. I componenti che attraversano questi separatori di isolamento, quali accoppiatori ottici, condensatori, trasformatori e relè sono stati approvati come componenti che garantiscono una separazione sicura. Nelle schede tecniche dell'S7-1200 i separatori di isolamento che soddisfano tali requisiti sono identificati da un valore di isolamento di 1500 V AC o superiore. Questa definizione si basa su un test di routine di (2 Ue + 1000 V AC) o equivalente eseguito internamente in conformità a metodi approvati. I separatori di isolamento dell'S7-1200 sono stati testati a 4242 V DC.

L'uscita di alimentazione dei sensori, i circuiti di comunicazione e i circuiti logici interni di un S7-1200 con alimentazione AC integrata sono classificati come SELV (bassissima tensione di sicurezza) ai sensi della norma EN 61131-2.

Per mantenere il carattere di sicurezza dei circuiti a bassa tensione dell'S7-1200, le connessioni esterne alle porte di comunicazione, i circuiti analogici e tutti i circuiti I/O e di alimentazione con tensione nominale di 24 V devono essere alimentati da sorgenti approvate che soddisfano i requisiti di SELV, PELV, Classe 2, tensione limitata o potenza limitata a seconda della norma.

AVVERTENZA

Se si utilizzano alimentatori non isolati o a isolamento singolo per alimentare i circuiti con bassa tensione da una linea AC, si possono generare tensioni pericolose nei circuiti che dovrebbero invece essere sicuri in caso di contatto dell'operatore, come i circuiti di comunicazione e il cablaggio a bassa tensione dei sensori.

Queste alte tensioni improvvise possono provocare scosse elettriche e causare la morte o gravi lesioni alle persone e/o danni alle cose.

Utilizzare esclusivamente convertitori da alta a bassa tensione approvati come sorgente di alimentazione di circuiti a tensione limitata sicura in caso di contatto dell'operatore.

Istruzioni per la messa a terra dell'S7-1200

Il modo migliore per mettere a terra la propria applicazione è di garantire che tutte le connessioni comuni e di terra dell'S7-1200 e delle relative apparecchiature siano collegate ad un unico punto di massa. Tale punto comune va connesso direttamente alla terra del sistema.

I conduttori di terra devono essere più corti possibile e avere un diametro elevato, ad es. di 2 mm² (14 AWG).

Quando si progettano i punti di collegamento a terra si deve inoltre tener conto dei requisiti di sicurezza e accertarsi del corretto funzionamento dei dispositivi di protezione di interruzione del circuito.

Istruzioni per il cablaggio dell'S7-1200

Quando si progetta il cablaggio dell'S7-1200 si deve prevedere un unico interruttore che disinserisca contemporaneamente la corrente dall'alimentatore della CPU S7-1200, dai circuiti di ingresso e da quelli di uscita. Installare un dispositivo di protezione dalla sovracorrente, ad es. un fusibile o un interruttore automatico, che limiti le correnti anomale nel cablaggio di alimentazione. Valutare se non sia opportuno installare un fusibile o un altro limitatore di corrente in ciascun circuito di uscita per ottenere una protezione ancora maggiore.

Per i conduttori che possono essere soggetti a sovratensioni dovute ai fulmini si devono prevedere appositi dispositivi di soppressione delle sovratensioni.

È importante non disporre i conduttori di segnale a bassa tensione e i cavi di comunicazione assieme ai conduttori di potenza AC e ai conduttori DC ad alta corrente e a commutazione rapida. Posare sempre i conduttori a coppie: il neutro o filo comune con il filo caldo o filo di segnale.

Utilizzare un conduttore più corto possibile e verificare che abbia una sezione adatta alla corrente richiesta. La morsettieria accetta conduttori con dimensioni da 2 mm² a 0,3 mm² (da 14 AWG a 22 AWG). Utilizzare cavi schermati per migliorare la protezione dal rumore elettrico. Generalmente il risultato migliore si ottiene mettendo a terra lo schermo sull'S7-1200.

Se i circuiti di ingresso sono alimentati da un alimentatore esterno, inserire nel circuito una protezione dalla sovracorrente. La protezione esterna non è necessaria nei circuiti alimentati dall'alimentazione per sensori a 24 V DC dell'S7-1200, poiché questa è già limitata in corrente.

Tutte le unità S7-1200 sono dotate di morsettieria a innesto per il cablaggio. Per impedire che le connessioni si allentino, controllare che la morsettieria sia ben inserita e che il conduttore sia installato correttamente. Per non danneggiare la morsettieria non serrare eccessivamente le viti. Utilizzare una coppia di max. 0,56 N-m.

L'S7-1200 prevede separatori di isolamento in determinati punti per evitare che si formino flussi di corrente indesiderati nell'installazione. Quando si progetta il cablaggio del sistema si deve quindi tener conto della posizione di tali limiti di isolamento. Per maggiori informazioni sull'isolamento fornito e la collocazione dei separatori consultare i dati tecnici. I separatori di isolamento impostati a meno di 1.500 V AC non sono affidabili come limiti di sicurezza.

Istruzioni per i carichi induttivi

Per limitare l'innalzamento della tensione quando si disattiva l'uscita di controllo, si devono dotare i carichi induttivi di circuiti di soppressione. I circuiti di soppressione impediscono che le uscite si guastino prematuramente a causa delle tensioni elevate determinate dallo spegnimento dei carichi induttivi. Inoltre, essi limitano il rumore elettrico che si genera quando vengono commutati i carichi induttivi. Il sistema più efficace per ridurre il rumore elettrico consiste nell'utilizzare un circuito di soppressione esterno collocandolo in modo che sia elettricamente in parallelo al carico e fisicamente vicino ad esso.

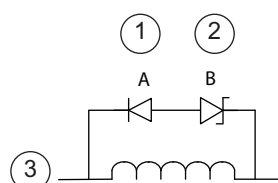
Nota

L'efficacia di un circuito di soppressione dipende dall'applicazione e deve essere verificata per ogni caso specifico. Controllare sempre che i componenti del circuito di soppressione siano impostati in modo adeguato all'applicazione.

Controllo di carichi induttivi DC

Le uscite DC dell'S7-1200 sono dotate di circuiti di soppressione adeguati ai carichi induttivi della maggior parte delle applicazioni. Poiché i relè possono essere utilizzati per un carico sia DC che AC, non è prevista alcuna protezione interna. La seguente figura illustra un esempio di circuito di soppressione per un carico DC.

Nella maggior parte delle applicazioni è opportuno utilizzare un diodo (A) in parallelo al carico induttivo, ma se l'applicazione richiede tempi di disattivazione più rapidi, è consigliabile utilizzare un diodo Zener (B).

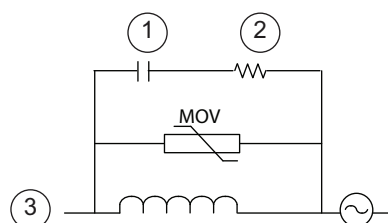


- ① Diodo 1N4001 o equivalente
- ② Zener da 8.2 V (uscite DC), Zener da 36 V (uscite relè)
- ③ Uscita

Verificare che la potenza del diodo Zener sia adeguata alla quantità di corrente del circuito di uscita.

Uscite relè che controllano i carichi AC

Se si utilizza un'uscita relè per commutare i carichi da 115 V/230 V AC, collocare una resistenza/un condensatore in parallelo al carico AC come indicato nella figura. Si può inoltre inserire un varistore a ossido metallico (MOV) per limitare la tensione di picco. Accertarsi che la tensione di lavoro del varistore sia almeno del 20% superiore alla tensione nominale di linea.



- ① 0,1 μ F
- ② Da 100 a 120 Ω
- ③ Uscita

Istruzioni per i carichi delle lampade

I carichi delle lampade possono danneggiare i contatti dei relè a causa dell'elevata sovracorrente transitoria all'accensione, la quale può essere da 10 a 15 volte superiore alla corrente di funzionamento di una lampadina con filamento al tungsteno. Per i carichi lampade che vengono commutati molto frequentemente durante il ciclo di vita dell'applicazione, si consiglia pertanto di utilizzare un relè sostituibile o un limitatore di corrente.

Concetti base sui PLC

3.1 Esecuzione del programma utente

La CPU supporta i seguenti tipi di blocchi di codice che consentono di creare una struttura efficiente per il programma utente:

- I blocchi organizzativi (OB) definiscono la struttura del programma. In alcuni OB il comportamento e gli eventi di avvio sono predefiniti, ma l'utente può creare anche OB con eventi di avvio a sua scelta.
- Le funzioni (FC) e i blocchi funzionali (FB) contengono il codice del programma che corrisponde a specifici task o combinazioni di parametri. Ogni FC o FB dispone di un set di parametri di ingresso e di uscita per condividere i dati con il blocco richiamante. Gli FB sono inoltre associati a blocchi dati (detti "DB di istanza") che ne mantengono i valori tra un'esecuzione e l'altra, in modo da consentirne l'uso in altri blocchi del programma.
- I blocchi dati (DB) memorizzano i dati utilizzabili nei blocchi di programma.

L'esecuzione del programma utente inizia con l'esecuzione di uno o alcuni OB di avvio in seguito all'attivazione del modo RUN e prosegue con l'esecuzione ciclica di uno o alcuni OB di ciclo. Gli OB possono essere associati a eventi di allarme (standard o di errore) e vengono eseguiti quando si verifica l'evento corrispondente.

Le funzioni (FC) e i blocchi funzionali (FB) sono blocchi del codice di programma che possono essere richiamati da un OB o da un'altra FC o un altro FB, fino a raggiungere i seguenti livelli:

- 16 dall'OB di ciclo o di avvio del programma
- 4 dagli OB di allarme di ritardo, di schedulazione orologio, di interrupt di processo, di errore temporale o di allarme di diagnostica

Le FC non sono associate a un blocco dati (DB) particolare, mentre gli FB sono collegati direttamente a un DB di cui si servono per passare parametri e memorizzare valori e risultati intermedi.

Le dimensioni del programma utente, dei dati e della configurazione sono limitate dalla memoria di caricamento disponibile e dalla memoria di lavoro nella CPU. Non c'è alcun limite rispetto al numero di blocchi supportato dalla quantità di memoria di lavoro disponibile.

Ciascun ciclo comprende la scrittura della uscite, la lettura degli ingressi, l'esecuzione delle istruzioni del programma utente, la manutenzione del sistema o l'elaborazione in background. Il ciclo viene chiamato "ciclo di scansione" o "scansione".

Le signal board, i moduli di I/O e i moduli di comunicazione vengono rilevati e collegati solo all'accensione.

Nota

Non possono essere inseriti ed estratti quando l'alimentazione è collegata. Fa eccezione solo la memory card SIMATIC che può essere inserita ed estratta anche quando la CPU è accesa.

Nella configurazione di default tutti gli I/O digitali e analogici vengono aggiornati in modo sincrono rispetto al ciclo di scansione mediante un'area di memoria interna chiamata "immagine di processo". L'immagine di processo rispecchia la situazione attuale degli ingressi e delle uscite fisici (quelli della CPU, della signal board e dei moduli di I/O).

La CPU esegue i seguenti task:

- La CPU scrive nelle uscite fisiche i valori letti dall'area dell'immagine di processo delle uscite.
- Legge gli ingressi fisici immediatamente prima dell'esecuzione del programma utente e ne memorizza i valori nell'area dell'immagine di processo degli ingressi. In questo modo i valori rimangono coerenti durante l'esecuzione delle istruzioni utente.
- Esegue la logica delle istruzioni utente e aggiorna i valori di uscita nell'area dell'immagine di processo delle uscite invece di scriverli nelle uscite fisiche.

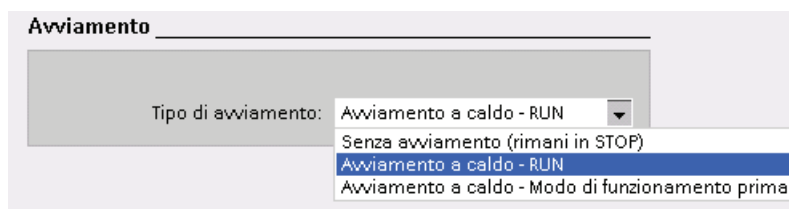
Questo processo garantisce la coerenza della logica durante l'esecuzione delle istruzioni utente per un dato ciclo e impedisce l'instabilità delle uscite che potrebbero cambiare stato più volte nell'area dell'immagine di processo delle uscite.

Si può specificare se gli I/O digitali e analogici devono essere salvati nell'immagine di processo. Se si inserisce il modulo nella vista dispositivi i relativi dati vengono collocati nell'immagine di processo della CPU S7-1200 (default). La CPU gestisce automaticamente lo scambio dei dati tra il modulo e l'area dell'immagine di processo durante l'aggiornamento dell'immagine di processo. Per eliminare degli I/O digitali o analogici dall'aggiornamento automatico dell'immagine di processo, selezionare il dispositivo appropriato in Configurazione dispositivi, visualizzare la scheda Proprietà, se necessario espanderla per individuare gli I/O desiderati e quindi selezionare "Identificatore HW/Indirizzi I/O". Modificare la voce di "Immagine di processo:" da "IP ciclico" in "---". Per reintegrare gli I/O nell'aggiornamento automatico dell'immagine di processo ripristinare l'impostazione "IP ciclico".

Mentre l'istruzione viene eseguita si possono leggere direttamente i valori degli ingressi fisici e scrivere direttamente quelli delle uscite. La lettura diretta accede allo stato attuale dell'ingresso fisico senza aggiornare l'area dell'immagine di processo, indipendentemente dal fatto che l'ingresso sia configurato per essere salvato nell'immagine di processo. La scrittura diretta in un'uscita fisica aggiorna sia l'area dell'immagine di processo delle uscite (se l'uscita è configurata per essere salvata nell'immagine di processo) che l'uscita fisica. Per fare in modo che il programma acceda ai dati di I/O direttamente da un ingresso o un'uscita fisica invece che dall'immagine di processo aggiungere il suffisso ":P" all'indirizzo di I/O.

Configurazione dei parametri di avvio

Per definire come la CPU si avvierà dopo un ciclo di spegnimento/riaccensione si utilizzano le proprietà della CPU.



Selezionare se la CPU si avvierà in STOP, in RUN o con il modo di funzionamento precedente al ciclo di spegnimento/riaccensione.

La CPU effettua un avviamento a caldo prima di passare in RUN. L'avviamento a caldo reimposta la memoria non a ritenzione sui valori iniziali di default, ma mantiene i valori presenti nella memoria a ritenzione.

Nota

Dopo un caricamento la CPU effettua sempre un avviamento

Quando si carica un elemento del progetto (ad es. un blocco di programma, un blocco dati o una configurazione hardware) la CPU effettua un avviamento al successivo passaggio in RUN. Oltre ad azzerare gli ingressi e a inizializzare le uscite e la memoria non a ritenzione, l'avviamento cancella anche le aree di memoria a ritenzione.

Al termine dell'avviamento che segue un caricamento, tutti i successivi passaggi da STOP a RUN determinano un avviamento a caldo (che non inizializza la memoria a ritenzione).

3.1.1 Modi di funzionamento della CPU

La CPU prevede tre modi di funzionamento: STOP, AVVIAMENTO e RUN. La modalità attiva è indicata dai LED di stato posti sul lato anteriore della CPU.

- Nel modo STOP la CPU non esegue il programma ed è possibile caricarvi un progetto.
- Nel modo AVVIAMENTO vengono eseguiti una volta gli OB di avvio (se presenti). Gli eventi di allarme non vengono elaborati durante la fase di avviamento del modo RUN.
- Nel modo RUN il ciclo di scansione viene eseguito ripetutamente. Gli eventi di allarme possono verificarsi ed essere elaborati in qualsiasi punto della scansione ciclica del programma.

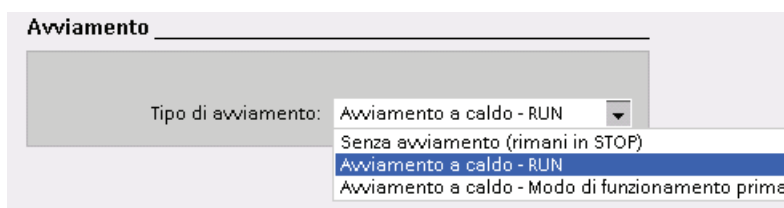
In RUN non è possibile caricare un progetto nella CPU.

La CPU supporta l'avviamento a caldo per passare in RUN. L'avviamento a caldo non implica il reset della memoria. Durante l'avviamento a caldo vengono inizializzati tutti i dati di sistema e i dati utente non a ritenzione. I dati utente a ritenzione rimangono in memoria.

L'operazione di reset cancella la memoria di lavoro e le aree a ritenzione e non a ritenzione e copia la memoria di caricamento in quella di lavoro. Non cancella invece il buffer di diagnostica o i valori dell'indirizzo IP memorizzati in modo permanente.

Il modo di funzionamento della CPU all'accensione e il metodo di riavviamento possono essere specificati con il software di programmazione. Questa opzione è disponibile in Avviamento nella finestra Configurazione dispositivi della CPU. All'accensione la CPU esegue una serie di test diagnostici e inizializza il sistema. Quindi passa nel modo di funzionamento appropriato. Alcuni errori impediscono alla CPU di passare in RUN. La CPU supporta i seguenti modi di funzionamento all'accensione:

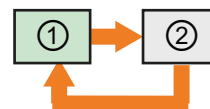
- Modo STOP
- Passaggio in RUN dopo un avviamento a caldo
- Passaggio al precedente modo di funzionamento dopo un avviamento a caldo



Per modificare il modo di funzionamento attuale si utilizzano i comandi "STOP" o "RUN" dei tool online del software di programmazione. È inoltre possibile portare la CPU in STOP inserendo nel programma un'istruzione STP, in modo che l'esecuzione venga arrestata in base alla logica del programma stesso.

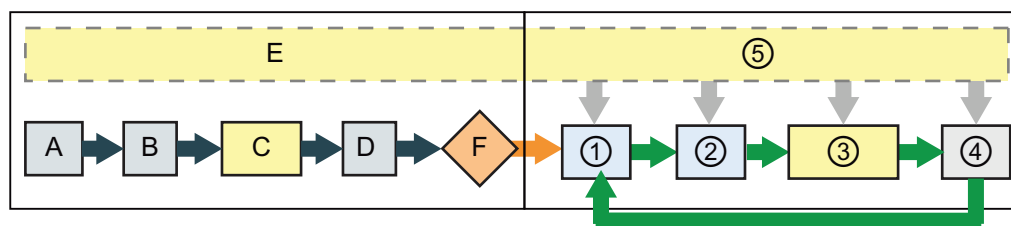
In STOP la CPU ① gestisce le richieste di comunicazione (come richiesto) ed ② esegue un autotest.

Non esegue invece il programma utente e l'immagine di processo non viene aggiornata automaticamente.



È possibile caricare il progetto solo quando la CPU è in STOP.

In RUN la CPU esegue i task indicati nella seguente figura.



AVVIAMENTO

- A Cancella l'area di memoria I
- B Inizializza le uscite con l'ultimo valore o con il valore sostitutivo
- C Esegue gli OB di avvio
- D Copia lo stato degli ingressi fisici nella memoria I
- E Memorizza gli eventi di allarme nella coda d'attesa da elaborare in RUN
- F Abilita la scrittura della memoria Q nelle uscite fisiche

RUN

- ① Scrive la memoria Q nelle uscite fisiche
- ② Copia lo stato degli ingressi fisici nella memoria I
- ③ Esegue gli OB di ciclo
- ④ Esegue l'autotest
- ⑤ Elabora gli allarmi e la comunicazione in un punto qualsiasi del ciclo di scansione

Elaborazione dell'AVVIAMENTO

Quando lo stato di funzionamento cambia da STOP a RUN, la CPU azzerà gli ingressi dell'immagine di processo, inizializza l'immagine di processo delle uscite ed elabora l'OB di avvio. Ogni volta che le istruzioni dell'OB di avvio accedono in lettura agli ingressi dell'immagine di processo, leggono uno zero invece del valore attuale degli ingressi fisici. Per leggere lo stato attuale di un ingresso fisico durante il modo di avviamento si deve effettuare una lettura diretta. In seguito vengono eseguiti gli OB di avvio e le FC e gli FB a cui sono associati. Se sono presenti più OB di avvio vengono eseguiti ciascuno in base al proprio numero, a partire da quello più basso.

Ogni OB di avvio include informazioni che consentono di determinare se i dati a ritenzione e l'orologio hardware sono validi. All'interno degli OB di avvio si possono programmare delle istruzioni che esaminino tali valori di avvio ed eseguano l'azione appropriata. Gli OB di avvio supportano le seguenti posizioni di avvio:

Ingresso	Tipo di dati	Descrizione
LostRetentive	BOOL	Questo bit è vero se le aree di salvataggio dei dati a ritenzione sono state eliminate
LostRTC	BOOL	Questo bit è vero se l'orologio hardware (Real Time Clock) è stato eliminato

Durante l'elaborazione dell'avviamento la CPU esegue inoltre i task descritti di seguito.

- Gli allarmi vengono inseriti nella coda d'attesa ma non elaborati.
- Durante l'avviamento il tempo di ciclo non viene controllato.
- Durante l'avviamento è possibile modificare la configurazione dei moduli HSC (contatori veloci), PWM (modulazione dell'ampiezza degli impulsi) e PtP (comunicazione punto a punto).
- Il funzionamento effettivo dei moduli HSC, PWM e di comunicazione punto a punto viene attivato solo in RUN.

Al termine dell'esecuzione degli OB di avvio la CPU passa in RUN ed elabora i task di comando in un ciclo di scansione continuo.

Elaborazione del ciclo di scansione in RUN

In tutti i cicli di scansione la CPU scrive nelle uscite, legge gli ingressi, esegue il programma utente, aggiorna i moduli di comunicazione, esegue una manutenzione interna e reagisce agli eventi e alle condizioni di allarme dell'utente. Le richieste di comunicazione vengono gestite periodicamente nel corso del ciclo di scansione.

Queste operazioni (fatta eccezione per gli eventi di allarme utente) vengono eseguite regolarmente e in sequenza. Gli eventi di allarme utente abilitati vengono elaborati in base alla loro priorità nell'ordine in cui si verificano.

Il sistema garantisce che il ciclo di scansione termini entro un tempo detto "tempo di controllo del ciclo" o "tempo massimo"; in caso contrario viene generato un evento di errore temporale.

- Ciascun ciclo di scansione inizia recuperando dall'immagine di processo i valori attuali delle uscite digitali e analogiche e scrivendoli nelle uscite fisiche della CPU e dei moduli SB e SM configurati per l'aggiornamento automatico degli I/O (configurazione di default). Quando un'istruzione accede a un'uscita fisica vengono aggiornate sia l'immagine di processo delle uscite che l'uscita fisica stessa.
- Il ciclo di scansione continua leggendo i valori attuali degli ingressi digitali e analogici dalla CPU e dai moduli SB e SM configurati per l'aggiornamento automatico (configurazione di default) e scrivendoli nell'immagine di processo. Quando un'istruzione accede a un ingresso fisico, accede al suo valore ma non aggiorna l'immagine di processo degli ingressi.
- Dopo la lettura degli ingressi il programma utente viene eseguito dalla prima all'ultima istruzione, ovvero vengono elaborati tutti gli OB di ciclo e le FC e gli FB a cui sono associati. Gli OB di ciclo vengono eseguiti in base al loro numero, a partire da quello più basso.

La comunicazione viene elaborata periodicamente nel corso del ciclo di scansione, eventualmente interrompendo l'esecuzione del programma utente.

L'autotest comprende controlli periodici del sistema e dello stato dei moduli di I/O.

Gli allarmi possono verificarsi durante una parte qualsiasi del ciclo di scansione e sono comandati da eventi. Quando si verifica un evento la CPU interrompe il ciclo di scansione e richiama l'OB configurato per elaborarlo. Quando l'OB conclude l'elaborazione dell'evento la CPU riprende l'esecuzione del programma utente dal punto in cui l'ha interrotto.

Blocchi organizzativi (OB)

Gli OB controllano l'esecuzione del programma utente. Ogni OB deve essere contrassegnato da un numero univoco. Alcuni numeri di default inferiori a 200 sono riservati. Gli altri OB devono essere numerati da 200 in poi.

Alcuni eventi specifici della CPU attivano l'esecuzione di un blocco organizzativo. Gli OB non possono richiamarsi tra loro, né essere richiamati da un'FC o un FB. L'esecuzione di un OB può essere avviata solo da un evento di avvio, ad es. da un allarme di diagnostica o da un intervallo di tempo. La CPU gestisce gli OB in base alla loro classe di priorità, eseguendo per primi quelli con priorità superiore e in seguito quelli con priorità inferiore. La classe di priorità più bassa è la 1 (per il ciclo di programma principale) e quella più alta la 27 (per gli allarmi di errore temporale).

Gli OB controllano le seguenti operazioni:

- Gli OB di ciclo vengono eseguiti ciclicamente quando la CPU è in RUN. Il blocco principale del programma è un OB di ciclo che contiene le istruzioni che controllano il programma e i richiami degli altri blocchi utente. È consentito utilizzare più OB di ciclo ed eseguirli in ordine numerico. Per default è impostato l'OB 1. Gli altri OB di ciclo devono essere identificati come OB 200 o superiore.
- Gli OB di avvio vengono eseguiti una volta quando il modo di funzionamento della CPU cambia da STOP a RUN, anche quando la CPU passa in RUN all'accensione e nelle transizioni da STOP a RUN attivate da un comando. Quando termina l'avvio inizia l'esecuzione dell'OB di ciclo principale. È possibile utilizzare più OB di avvio. Per default è impostato l'OB 100. Gli altri devono avere il numero 200 o un numero superiore.
- Gli OB di allarme di ritardo vengono eseguiti a intervalli specifici dopo che l'istruzione Allarme di avvio (SRT_DINT) ha configurato un evento. Il tempo di ritardo è specificato nel parametro di ingresso dell'istruzione ampliata SRT_DINT. Gli OB di allarme di ritardo interrompono la normale esecuzione ciclica del programma al termine di un tempo di ritardo specificato. Si possono configurare fino a 4 eventi di ritardo in qualsiasi momento, utilizzando un OB per ciascuno di essi. Gli OB di allarme di ritardo devono avere il numero 200 o un numero superiore.
- Gli OB di schedulazione orologio vengono eseguiti a un intervallo specificato. Essi interrompono l'esecuzione ciclica del programma a intervalli definiti dall'utente, ad es. ogni 2 secondi. Si possono configurare fino a 4 eventi di schedulazione orologio, utilizzando un OB per ognuno di essi. L'OB deve avere il numero 200 o un numero superiore.
- Gli OB di interrupt di processo vengono eseguiti quando si verifica l'evento di processo rilevante, ad es. un fronte di salita e di discesa negli ingressi integrati o un evento HSC. Gli OB di interrupt di processo interrompono la normale esecuzione ciclica del programma in risposta al segnale proveniente da un evento di processo. Gli eventi possono essere definiti nelle proprietà della configurazione hardware. È consentito utilizzare un OB per ciascun evento di processo configurato. L'OB deve avere il numero 200 o un numero superiore.
- Gli OB di errore temporale vengono eseguiti quando viene rilevato un errore temporale. Gli OB di errore temporale interrompono la normale esecuzione ciclica del programma se viene superato il tempo di controllo del ciclo definito nelle proprietà del PLC. L'OB 80 è l'unico numero di OB consentito per l'evento di errore temporale. È possibile configurare l'azione che deve essere eseguita quando l'OB 80 non è presente: ignorare l'errore o passare in STOP.
- Gli OB di allarme di diagnostica vengono eseguiti quando viene rilevato e segnalato un errore di diagnostica. Gli OB di allarme di diagnostica interrompono la normale esecuzione ciclica del programma se un modulo con funzioni di diagnostica rileva un errore (deve essere stato abilitato l'allarme per errore di diagnostica per il modulo). L'OB 82 è l'unico numero di OB consentito per l'evento di errore di diagnostica. Se il programma non contiene OB di diagnostica si può configurare la CPU in modo che ignori l'errore o passi in STOP.

3.1.2 Priorità di esecuzione degli eventi e loro inserimento nella coda d'attesa

L'elaborazione della CPU è comandata da eventi i quali attivano gli OB di allarme da eseguire. L'OB di allarme per un dato evento può essere specificato quando si creano il blocco o la configurazione dei dispositivi oppure mediante un'istruzione ATTACH o DETACH. Alcuni eventi, come quelli ciclici o del ciclo del programma, si verificano con regolarità, altri, ad es. gli eventi di avvio e di ritardo, una sola volta. Alcuni eventi, come quelli di fronte negli ingressi o dei contatori veloci, si verificano in seguito a un cambiamento determinato dall'hardware. Vi sono inoltre eventi, come quelli di errore di diagnostica e errore temporale, che si verificano solo in seguito a un errore. Per determinare l'ordine di elaborazione degli OB di allarme si utilizzano le priorità, i gruppi di priorità e le code.

L'evento di ciclo del programma si verifica una volta durante il ciclo (la scansione) del programma. Durante il ciclo del programma la CPU scrive le uscite, legge gli ingressi ed esegue gli OB di ciclo. L'evento di ciclo è indispensabile e sempre abilitato. Per tale evento può non essere disponibile alcun OB di ciclo o possono essere stati selezionati più OB. Una volta attivato l'evento di ciclo, viene eseguito l'OB di ciclo con il numero inferiore (solitamente l'OB1). Gli altri OB di ciclo vengono eseguiti in sequenza, in ordine numerico, entro il ciclo del programma.

Gli eventi di schedulazione orologio consentono di configurare l'esecuzione di un OB di allarme a un intervallo di tempo configurato. L'intervallo può essere configurato quando si crea l'OB e lo si imposta come OB di schedulazione orologio. Gli eventi di schedulazione interrompono il ciclo del programma ed eseguono l'OB di schedulazione orologio (l'evento si trova in un gruppo di priorità superiore a quello del ciclo di programma). Ad un OB di schedulazione orologio è possibile assegnare un solo evento di schedulazione orologio. La CPU supporta quattro eventi di questo tipo. Gli OB di schedulazione orologio determinano uno spostamento di fase, per cui l'esecuzione degli allarmi di schedulazione con lo stesso periodo può essere spostata di un dato tempo.

L'evento di avvio si verifica una volta in seguito al passaggio da STOP a RUN e determina l'esecuzione degli OB di avvio. Si possono selezionare diversi OB per l'evento di avvio, che vengono eseguiti in ordine numerico.

Gli eventi di allarme di ritardo consentono di configurare l'esecuzione di un OB di allarme dopo che è trascorso il tempo di ritardo specificato. Per specificarlo si utilizza l'istruzione SRT_DINT. Gli eventi di ritardo interrompono il ciclo del programma per eseguire l'OB di allarme di ritardo. È possibile assegnare un solo OB di ritardo a un evento di ritardo. La CPU supporta quattro eventi di questo tipo.

Gli eventi di interrupt di processo vengono attivati da un cambiamento dell'hardware, ad es. un fronte di salita o di discesa in un ingresso o un evento HSC (contatore veloce). Per ciascun evento di interrupt di processo si può selezionare un unico OB di allarme. Gli eventi di interrupt di processo vengono abilitati nella finestra Configurazione dispositivi. Gli OB per l'evento possono essere specificati nella finestra Configurazione dispositivi o con un'istruzione ATTACH del programma utente. La CPU supporta diversi eventi di interrupt di processo. Gli eventi precisi dipendono dal modello di CPU e dal numero di ingressi.

Gli eventi di errore temporale o di allarme di diagnostica vengono attivati quando la CPU rileva un errore. Appartengono a un gruppo di priorità superiore rispetto agli altri eventi di allarme e possono interrompere l'esecuzione degli eventi di allarme di ritardo, di schedulazione orologio e di interrupt di processo. Si può specificare un solo OB di allarme per ciascun evento di errore temporale e di allarme di diagnostica.

Priorità di esecuzione degli eventi e loro inserimento nella coda d'attesa

Il numero di eventi in attesa (messi in coda) provenienti da un'unica origine può essere limitato utilizzando una coda diversa per ciascun tipo di evento. Quando viene raggiunto il limite previsto per un dato tipo di eventi, l'evento successivo viene eliminato. Per maggiori informazioni sugli overflow della coda d'attesa consultare il capitolo "Eventi di errore temporale".

Ogni evento della CPU è associato a una priorità e le priorità sono suddivise in classi. La seguente tabella riepiloga le profondità delle code, le classi di priorità e le priorità degli eventi della CPU supportati.

Nota

La priorità, la classe di priorità e le profondità delle code non sono modificabili.

In generale gli eventi vengono elaborati in base alla loro priorità (iniziando da quella maggiore). Gli eventi con la stessa priorità vengono elaborati in base all'ordine di arrivo.

Tipo di evento (OB)	Quantità	Numeri di OB validi	Profondità coda	Classe di priorità	Priorità
Ciclo del programma	1 evento di ciclo del programma. Sono consentiti più OB.	1 (default) 200 o superiore.	1	1	1
Avviamento	1 evento di avvio ¹ . Sono consentiti più OB.	100 (default) 200 o superiore.	1		1
Allarme di ritardo	4 eventi di ritardo. 1 OB per evento.	200 o superiore.	8	2	3
Di ciclo	4 eventi ciclici. 1 OB per evento.	200 o superiore.	8		4
Fronti	16 eventi di fronte di salita. 16 eventi di fronte di discesa. 1 OB per evento.	200 o superiore.	32		5
HSC	6 eventi CV = PV. 6 eventi di cambio direzione. 6 eventi di reset esterno. 1 OB per evento.	200 o superiore.	16		6
Allarme di diagnostica	1 evento	Solo OB 82.	8		9
Evento di errore temporale / evento di tempo di ciclo max.	1 evento di errore temporale 1 evento di tempo di ciclo max.	Solo OB 80.	8	3	26

Tipo di evento (OB)	Quantità	Numeri di OB validi	Profondità coda	Classe di priorità	Priorità
2 eventi di tempo di ciclo max.	2 eventi di tempo di ciclo max.	Nessun OB richiamato	-	3	27
¹ Casi speciali per l'evento di avvio - L'evento di avvio e quello di ciclo del programma non possono verificarsi contemporaneamente perché il sistema operativo avvia l'evento di ciclo del programma dopo che è terminato quello di avvio. - L'evento di avvio non può essere interrotto da altri eventi. Gli eventi che si verificano durante l'evento di avvio vengono messi in coda per essere elaborati al termine dell'evento.					

Una volta avviata, l'elaborazione di un OB non può essere interrotta da altri eventi della stessa classe di priorità o di una classe inferiore. Tali eventi vengono messi in coda d'attesa ed elaborati in seguito, in modo che l'OB possa concludere la propria esecuzione.

Solo se si verifica un evento di una classe di priorità superiore l'OB in corso viene interrotto e la CPU esegue quello associato all'evento di priorità superiore. Quando termina l'esecuzione dell'OB di priorità superiore, la CPU esegue gli OB degli altri eventi che sono in coda nella stessa classe, in base alla loro rispettiva priorità. Se non sono presenti altri eventi in attesa per questa classe di priorità, la CPU torna alla classe inferiore e riprende l'elaborazione dell'OB interrotto dal punto in cui era stata interrotta.

Latenza degli allarmi

La latenza degli eventi di allarme (ovvero il tempo che trascorre dal momento in cui la CPU notifica che un evento si è verificato fino a quando la CPU inizia a eseguire la prima istruzione dell'OB di elaborazione dell'evento) è di circa 210 µsec, sempre che, quando si verifica l'evento, l'OB di ciclo del programma sia l'unica routine di elaborazione attiva.

Eventi di errore temporale

Gli eventi di errore temporale possono essere causati da una delle seguenti condizioni :

- Superamento del tempo di controllo del ciclo
- Impossibilità di avviare l'OB richiesto
- Overflow della coda d'attesa

La condizione di "superamento del tempo di controllo del ciclo" si verifica quando il ciclo del programma non termina entro il tempo di controllo specificato. Per maggiori informazioni su questa condizione, sulla configurazione del tempo di controllo del ciclo e sul reset del temporizzatore di ciclo si consiglia di consultare il capitolo "Controllo del tempo di ciclo (Pagina 48)".

La condizione di "impossibilità di avviare l'OB richiesto" si verifica se un allarme di schedulazione orologio o un allarme di ritardo richiedono un OB che è già in esecuzione.

La condizione di "overflow della coda d'attesa" si verifica se gli allarmi si verificano più rapidamente di quanto possano essere elaborati. Il numero di eventi in attesa può essere limitato utilizzando una coda diversa per ciascun tipo di evento. Se un evento si verifica quando la coda corrispondente è piena viene generato un evento di errore temporale.

Tutti gli eventi di errore temporale attivano l'esecuzione dell'OB 80 (se presente). Se l'OB 80 non esiste la CPU ignora l'errore. Se all'interno dello stesso ciclo si verificano due condizioni di superamento del tempo di controllo e il temporizzatore del ciclo non viene resettato, la CPU passa in STOP anche se l'OB 80 è presente. Consultare in proposito il capitolo "Controllo del tempo di ciclo". (Pagina 48)

L'OB 80 contiene informazioni di avvio che consentono di determinare quale evento e quale OB hanno generato l'errore temporale. All'interno dell'OB 80 si possono programmare delle istruzioni che esaminino tali valori di avvio ed eseguano l'azione appropriata. L'OB 80 supporta le seguenti posizioni di avvio:

Ingresso	Tipo di dati	Descrizione
fault_id	BYTE	16#01 - superamento del tempo di controllo del ciclo 16#02 - impossibilità di avviare l'OB richiesto 16#07 e 16#09 - overflow della coda d'attesa
csg_OBnr	OB_ANY	Numero dell'OB che era in esecuzione quando si è verificato l'errore
csg_prio	UINT	Priorità dell'OB che ha causato l'errore

Quando si crea un nuovo progetto l'OB 80 di errore temporale non è presente. Se lo si desidera lo si può inserire nel progetto facendo doppio clic su "Inserisci nuovo blocco" in "Blocchi di programma" nell'albero e scegliendo in seguito "Blocco organizzativo" e "OB di errore temporale".

Eventi di errore di diagnostica

Alcuni dispositivi sono in grado di rilevare e segnalare gli errori di diagnostica. Gli eventi di errore di diagnostica vengono generati in seguito al verificarsi o all'eliminazione di svariate condizioni di errore di diagnostica. Gli errori di diagnostica supportati sono:

- Mancanza di alimentazione
- Superamento del limite superiore
- Superamento del limite inferiore
- Rottura conduttore
- Cortocircuito

Tutti gli eventi di errore di diagnostica attivano l'esecuzione dell'OB 82 (se presente). Se l'OB 82 non esiste la CPU ignora l'errore. Quando si crea un nuovo progetto l'OB 82 di errore di diagnostica non è presente. Se lo si desidera lo si può inserire nel progetto facendo doppio clic su "Inserisci nuovo blocco" in "Blocchi di programma" nell'albero e scegliendo in seguito "Blocco organizzativo" e "Allarme di diagnostica".

L'OB 82 include informazioni che consentono di determinare se l'evento è causato dal verificarsi o dall'eliminazione di un errore e di stabilire il dispositivo o canale che lo hanno rilevato. All'interno dell'OB 82 si possono programmare delle istruzioni che esaminino tali valori di avvio ed eseguano l'azione appropriata. L'OB 82 supporta le seguenti posizioni di avvio:

Ingresso	Tipo di dati	Descrizione
IOstate	WORD	Stato di I/O del dispositivo
laddr	HW_ANY	Identificatore hardware del dispositivo o dell'unità funzionale che hanno rilevato l'errore
channel	UINT	Numero di canale
multierror	BOOL	TRUE se è presente più di un errore (<i>non supportato nelle release precedenti</i>)

Il bit 4 di IO_state indica se l'evento è dovuto al verificarsi o all'eliminazione di un errore. Il bit 4 è 1 se è presente un errore (esempio: rottura conduttore) e 0 se l'errore non è più presente.

L'ingresso KOP contiene l'identificatore hardware (HW ID) del dispositivo o dell'unità funzionale che hanno rilevato un errore. L'ID hardware viene assegnato automaticamente quando si inseriscono i componenti nella vista dispositivi o in quella di rete e compare nella scheda "Costanti" delle variabili PLC. Inoltre l'ID viene denominato automaticamente. Queste voci della scheda Costanti delle variabili PLC non sono modificabili.

Il numero di canale inizia da 0 per il primo ingresso (analogico o digitale) e da 64 per la prima uscita (analogica o digitale). Sono necessari diversi offset per distinguere gli ingressi dalle uscite nel caso il dispositivo li contenga entrambi. Se un errore riguarda l'intero dispositivo o unità funzionale, ad es. la mancanza di alimentazione, viene impostato il bit più significativo della parola per il numero di canale (n° 32768).

Controllo del tempo di ciclo

Il tempo di ciclo è il tempo impiegato dal sistema operativo della CPU per eseguire la fase ciclica del modo RUN. La CPU consente di utilizzare due metodi per il controllo del tempo di ciclo:

- Tempo di controllo del ciclo
- Tempo di ciclo minimo fisso

Il controllo del ciclo di scansione inizia quando termina l'evento di avvio. Questa funzione può essere configurata in "Tempo di ciclo" nella finestra "Configurazione dispositivi" della CPU.

La CPU controlla sempre il ciclo di scansione e reagisce se viene superato il relativo tempo di controllo. Se viene superato il tempo del ciclo di scansione, viene generato un errore che verrà gestito in uno dei seguenti modi:

- Se non è presente l'OB 80 di allarme di errore temporale, la CPU genera un errore e continua ad eseguire il programma utente.
- Se è presente l'OB 80 di errore temporale la CPU lo esegue

Per resettare il temporizzatore che misura il tempo di ciclo si può usare l'istruzione RE_TRIGR (Riattiva il controllo del tempo di ciclo). L'istruzione funziona tuttavia solo se eseguita in un OB di ciclo e viene ignorata se eseguita nell'OB 80. Se il tempo di controllo del ciclo viene superato due volte entro lo stesso ciclo del programma senza che nel frattempo venga eseguita l'istruzione RE_TRIGR, la CPU passa immediatamente in STOP. L'esecuzione ripetuta dell'istruzione RE_TRIGR può determinare un loop continuo o un ciclo di scansione molto lungo.

Generalmente il ciclo di scansione viene eseguito più rapidamente possibile e il ciclo successivo inizia non appena termina quello in corso. La durata del ciclo di scansione può variare da un ciclo all'altro in funzione del programma utente e dei task di comunicazione. Per eliminare queste variazioni è possibile impostare un tempo di ciclo minimo fisso per la CPU. Se questa funzione opzionale è attiva ed è stato specificato un tempo di ciclo minimo fisso in ms, la CPU mantiene il tempo minimo per il completamento di ciascun ciclo entro ± 1 ms.

Se la CPU conclude il ciclo di scansione normale in un tempo inferiore al tempo di ciclo minimo specificato, utilizza il tempo rimanente per eseguire la diagnostica in runtime e/o per elaborare le richieste di comunicazione. In questo modo la CPU impiega sempre un tempo fisso per concludere un ciclo di scansione.

Se la CPU non conclude il ciclo di scansione entro il tempo di ciclo minimo specificato, lo porta a termine normalmente (compresa l'elaborazione della comunicazione) evitando che il sistema reagisca al superamento del tempo di ciclo minimo. La seguente tabella definisce i campi e i valori di default per le funzioni di controllo del tempo di ciclo.

Tempo di ciclo	Campo (ms)	Default
Tempo di controllo del ciclo ¹	Da 1 a 6000	150 ms
Tempo di ciclo minimo fisso ²	Da 1 al tempo di controllo del ciclo	Disattivato

¹ Il tempo di controllo del ciclo è sempre abilitato e va impostato su un valore compreso tra 1 ms e 6000 ms. Per default è selezionato il valore 150 ms.

² Il tempo di ciclo minimo fisso è opzionale ed è disattivato per default. Se necessario si deve selezionare un tempo di ciclo compreso tra 1 ms e il tempo di controllo del ciclo.

Configurazione del tempo di ciclo e del carico di comunicazione

Nelle proprietà della CPU della finestra Configurazione dispositivi si possono configurare i seguenti parametri:

- Tempo di ciclo: consente di indicare un tempo di controllo del ciclo ed eventualmente di definire anche un tempo di ciclo minimo fisso.

- Carico di comunicazione: consente di configurare una percentuale di tempo da riservare ai task di comunicazione.

Per maggiori informazioni sul ciclo di scansione consultare il capitolo "Controllo del tempo di ciclo". (Pagina 48)

3.1.3 Memoria della CPU

Gestione della memoria

La CPU mette a disposizione le seguenti aree di memoria per il salvataggio del programma utente, dei dati e della configurazione:

- La memoria di caricamento è non volatile e viene utilizzata per salvare il programma utente, i dati e la configurazione. Prima di essere caricato nella CPU il progetto viene salvato nell'area della memoria di caricamento che può trovarsi in una memory card (se presente) o nella CPU. Il contenuto di questa area di memoria non volatile viene mantenuto anche se viene a mancare l'alimentazione. La memory card ha uno spazio di memoria superiore a quello della memoria integrata nella CPU.
- La memoria di lavoro è volatile e viene utilizzata per salvare alcuni elementi del progetto durante l'esecuzione del programma utente. La CPU copia alcuni elementi del progetto dalla memoria di caricamento in quella di lavoro. Il contenuto di questa area volatile viene cancellato in caso di interruzione dell'alimentazione e ripristinato dalla CPU quando l'alimentazione viene ristabilita.
- La memoria a ritenzione è non volatile e consente di archiviare una quantità limitata di valori della memoria di lavoro. Viene utilizzata per salvare i valori di locazioni di memoria utente selezionate in caso di mancanza di alimentazione. In caso di interruzione dell'alimentazione la CPU ha un tempo di mantenimento sufficiente a conservare i valori di un numero limitato di locazioni specificate. Tali valori a ritenzione vengono ripristinati quando torna l'alimentazione.

Per sapere come viene utilizzata la memoria nel progetto attuale fare clic con il tasto destro del mouse sulla CPU (o su uno dei suoi blocchi) e selezionare "Risorse" nel menu a comparsa. Per visualizzare l'utilizzo della memoria nella CPU attuale cliccare due volte su "Online e diagnostica", espandere "Diagnostica" e selezionare "Memoria".

Memoria a ritenzione

Per evitare che particolari dati vadano persi in caso di interruzione dell'alimentazione, li si può impostare come dati a ritenzione. I dati configurabili come dati "a ritenzione" sono i seguenti:

- Memoria di merker (M): la larghezza precisa della memoria di merker può essere definita nella tabella delle variabili del PLC o nell'elenco delle assegnazioni. La memoria di merker a ritenzione inizia sempre dall'MB0 e prosegue senza interruzioni verso l'alto per il numero di byte specificato. Il valore può essere indicato nella tabella delle variabili del PLC o nell'elenco delle assegnazioni facendo clic sull'icona della barra degli strumenti "Ritenzione". Immettere il numero di byte M a ritenzione a partire da MB0.
- Variabili di un blocco funzionale (FB): se un FB è stato impostato come FB "Indirizzabile soltanto simbolicamente", il relativo editor di interfaccia contiene la colonna "Retain", che consente di selezionare "Retain" o "Non-Retain" per ciascuna variabile. Anche il DB di istanza creato in seguito all'inserimento dell'FB nell'editor di programma contiene questa colonna, ma solo per la visualizzazione; non è quindi possibile modificare lo stato "a ritenzione" dall'editor di interfaccia del DB di un FB che è stato configurato per essere "Indirizzabile soltanto simbolicamente".

Se un FB è stato creato senza selezionare la casella "Indirizzabile soltanto simbolicamente", il relativo editor di interfaccia non contiene la colonna "Retain". Il DB di istanza creato in seguito all'inserimento di questo FB nell'editor di programma contiene una colonna "Retain" modificabile. In questo caso se si seleziona l'opzione "Retain" per una variabile qualsiasi, vengono selezionate **tutte** le variabili. Allo stesso modo, se si deseleziona l'opzione per una variabile vengono deselezionate **tutte** le variabili. Se durante la configurazione di un FB è stata **disattivata** l'opzione "Indirizzabile soltanto simbolicamente", è possibile modificare lo stato "a ritenzione" dall'editor dei DB di istanza, sempre tenendo conto del fatto che tutte le variabili verranno impostate sullo stesso stato.

Una volta creato l'FB non è più possibile modificare l'opzione "Indirizzabile soltanto simbolicamente". La si può selezionare solo quando si crea l'FB. Per determinare se un FB è stato configurato come "Indirizzabile soltanto simbolicamente", selezionarlo e fare clic con il tasto destro del mouse nell'albero del progetto, scegliere "Proprietà" e quindi "Attributi".

- Variabili di un blocco dati globale: riguardo all'assegnazione dello stato "a ritenzione" i DB globali si comportano in modo simile agli FB. A seconda dell'impostazione per l'indirizzamento simbolico è possibile impostare lo stato "a ritenzione" per variabili singole o per tutte le variabili di un blocco dati globale.
 - Se si seleziona l'attributo "Indirizzabile soltanto simbolicamente" del DB è possibile impostare lo stato "a ritenzione" per ogni singola variabile.
 - Se non lo si seleziona, lo stato "a ritenzione" viene impostato per tutte le variabili del DB, per cui le variabili sono tutte a ritenzione oppure tutte non a ritenzione.

Complessivamente si possono avere 2048 byte di dati a ritenzione. Per sapere quanti dati a ritenzione sono disponibili fare clic sull'icona della barra degli strumenti "Ritenzione" nella tabella delle variabili del PLC o nell'elenco delle assegnazioni. Nonostante qui venga specificato il campo a ritenzione della memoria M, la seconda riga indica la memoria complessivamente disponibile sia per i merker che per i DB.

Buffer di diagnostica

La CPU dispone di un buffer di diagnostica che contiene una voce per ciascun evento di diagnostica. Ogni voce comprende la data e l'ora in cui si è verificato l'evento, la categoria e una descrizione dell'evento. Le voci sono visualizzate in ordine cronologico e l'evento più recente compare per primo. In presenza di alimentazione nella CPU questo log può contenere gli ultimi 50 eventi. Quando il log è pieno, un eventuale nuovo evento sostituisce quello meno recente. Se viene a mancare l'alimentazione vengono salvati gli ultimi dieci eventi.

Nel buffer di diagnostica vengono registrati i seguenti tipi di eventi:

- Gli eventi di diagnostica del sistema; ad esempio gli errori della CPU e dei moduli
- I cambiamenti dello stato della CPU (le accensioni, le commutazioni in STOP e in RUN)

Per accedere al buffer di diagnostica è necessario essere online. Il log si trova in "Online e diagnostica / Diagnostica / Buffer di diagnostica". Per maggiori informazioni sulla soluzione dei problemi e il test consultare il capitolo "Online e diagnostica".

Orologio hardware

La CPU dispone di un orologio hardware. Un condensatore ad elevata capacità fornisce l'energia necessaria per far funzionare l'orologio quando la CPU è spenta. Il condensatore si ricarica quando la CPU è alimentata e dopo un minimo di 2 ore generalmente ha energia sufficiente a far funzionare l'orologio per dieci giorni.

L'orologio hardware viene impostato sull'ora di sistema ovvero sul Coordinated Universal Time (UTC). STEP 7 Basic imposta l'orologio hardware sull'ora di sistema. Sono disponibili istruzioni che consentono di leggere l'ora di sistema (RD_SYS_T) o l'ora locale (RD_LOC_T). L'ora locale viene calcolata sulla base del fuso orario e degli offset per l'ora legale impostati nella configurazione dell'orologio della CPU.

L'orologio hardware della CPU può essere configurato nella proprietà "Ora". È inoltre possibile abilitare l'ora legale e specificarne l'inizio e la fine. Per poter impostare l'orologio hardware si deve essere online e aver aperto la vista "Online e diagnostica" della CPU. Utilizzare la funzione "Imposta data e ora".

Merker di sistema e di clock

I byte per i "merker di sistema" e i "merker di clock" possono essere abilitati nelle proprietà della CPU. La logica del programma può indirizzare i singoli bit di queste funzioni.

- È possibile assegnare un byte di memoria M ai merker di sistema. Il byte di merker di sistema contiene i quattro seguenti bit che possono essere indirizzati dal programma utente:
 - Il bit "Sempre 0 (low)" è sempre impostato a 0.
 - Il bit "Sempre 1 (high)" è sempre impostato a 1.
 - "Diagramma di diagnostica modificato" viene impostato a 1 per un ciclo di scansione dopo che la CPU ha registrato un evento di diagnostica. Poiché la CPU non imposta il bit "Diagramma di diagnostica modificato" finché non termina la prima esecuzione degli OB di ciclo, il programma utente non può rilevare se si è verificata una modifica della diagnostica né durante l'esecuzione degli OB di avvio, né durante la prima esecuzione degli OB di ciclo.
 - Il bit "primo ciclo" viene impostato a 1 per il primo ciclo di scansione al termine dell'esecuzione dell'OB di avvio (al termine del primo ciclo, il bit di "primo ciclo" viene impostato a 0).
- È possibile assegnare un byte di memoria M ai merker di clock. Ogni bit del byte configurato come merker di clock genera un impulso ad onda quadra. Il byte di merker di clock fornisce 8 diverse frequenze comprese tra 0,5 Hz (lenta) e 10 Hz (veloce). Questi bit possono essere utilizzati come bit di comando, in particolare per le istruzioni con i fronti, per attivare delle azioni cicliche nel programma utente.

La CPU inizializza questi byte quando il modo di funzionamento passa da STOP a STARTUP. Durante i modi STARTUP e RUN i merker di clock cambiano in modo sincrono rispetto all'orologio della CPU.



CAUTELA

Se si sovrascrivono i bit dei merker di sistema o i merker di clock, i dati di queste funzioni possono danneggiarsi e causare il funzionamento errato del programma utente provocando danni alle apparecchiature e lesioni al personale.

Poiché sia i merker di clock che quelli di sistema non sono riservati nella memoria M, le istruzioni o comunicazioni possono scrivervi danneggiando i dati.

È quindi opportuno evitare di scrivere i dati in questi indirizzi, al fine di garantire la corretta esecuzione delle funzioni, e prevedere sempre un circuito di arresto d'emergenza per il processo o la macchina.

Bit del merker di sistema

☒ Attiva l'utilizzo del byte del merker di sistema

Indirizzo del byte del merker di sistema (MBx): **1**

Primo ciclo: M1.0

Diagramma di diagnostica modificato: M1.1

Sempre 1 (high): M1.2

Sempre 0 (low): M1.3

I merker di sistema configurano un byte che si attiva (valore = 1) nelle condizioni indicate di seguito.

- Primo ciclo: si attiva per il primo ciclo di scansione in RUN
- Diagramma di diagnostica modificato:
- Sempre 1 (high): Sempre attivo
- Sempre 0 (low): Sempre disattivato

Bit del merker di clock

☒ Attiva l'utilizzo del byte del merker di clock

Indirizzo del byte del merker di clock (MBx): **0**

Clock 10 Hz: M0.0

Clock 5 Hz: M0.1

Clock 2,5 Hz: M0.2

Clock 2 Hz: M0.3

Clock 1,25 Hz: M0.4

Clock 1 Hz: M0.5

Clock 0,625 Hz: M0.6

Clock 0,5 Hz: M0.7

I merker di clock configurano un byte che attiva e disattiva ciclicamente i singoli bit a intervalli prestabiliti.

I merker di clock generano ognuno un impulso ad onda quadra nel corrispondente bit di memoria M. Questi bit possono essere utilizzati come bit di comando, in particolare per le istruzioni con i fronti, per attivare delle azioni cicliche nel codice utente.

Configurazione del comportamento dei valori di uscita quando la CPU è in STOP

È possibile configurare il comportamento che verrà assunto dalle uscite digitali e analogiche quando la CPU è in STOP. Per ogni uscita di una CPU o di un modulo SB o SM si può indicare se dovrà essere congelata sul suo valore o se dovrà utilizzare un valore sostitutivo:

- Sostituzione di un valore di uscita specifico (default): si specifica un valore sostitutivo per ogni (canale di) uscita della CPU o del modulo SB o SM.

Il valore sostitutivo di default per i canali di uscita digitali è OFF e quello per i canali di uscita analogici è 0.

- Congelamento delle uscite in modo che mantengano il loro ultimo stato: le uscite mantengono il valore che avevano quando si è verificata la commutazione da RUN a STOP. Dopo l'accensione le uscite vengono impostate sul valore sostitutivo di default.

Il comportamento delle uscite può essere configurato in Configurazione dispositivi. Selezionare i singoli dispositivi nella scheda "Proprietà" per configurarne le uscite.

Quando passa da RUN a STOP, la CPU mantiene l'immagine di processo e, in base alla configurazione, scrive i valori appropriati sia per le uscite digitali che per quelle analogiche.

3.1.4 Protezione mediante password della CPU S7-1200

La CPU prevede 3 livelli di protezione per limitare l'accesso a funzioni specifiche. Quando si configurano il livello di sicurezza e la password di una CPU si limitano le funzioni e le aree di memoria accessibili senza password.

Per configurare la password procedere nel seguente modo:

1. Selezionare la CPU in "Configurazione dispositivi".
2. Selezionare la scheda "Proprietà" nella finestra di ispezione.
3. Selezionare la proprietà "Protezione" per impostare il livello di protezione e immettere la password.

Nella password si distingue tra caratteri maiuscoli e minuscoli.

Ogni livello consente di accedere ad alcune funzioni senza password. Per default la CPU non pone limiti all'accesso e non è protetta da password. Per limitare l'accesso a una CPU se ne devono configurare le proprietà e specificare la password.

La CPU continua a essere protetta anche se la password viene immessa attraverso una rete. Quando è protetta dalla password la CPU consente l'accesso illimitato a un solo utente per volta. La protezione mediante password non viene applicata all'esecuzione delle istruzioni del programma e delle funzioni di comunicazione. Immettendo la password corretta si può accedere a tutte le funzioni.

La comunicazione da PLC a PLC (mediante le istruzioni di comunicazione dei blocchi di codice) non viene limitata dal livello di sicurezza della CPU. Non sono previsti limiti neppure per le funzioni HMI.

Livello di sicurezza	Limitazioni dell'accesso
Senza protezione	Consente l'accesso completo senza password.
Protetto in scrittura	Consente l'accesso HMI e tutti i tipi di comunicazione da PLC a PLC senza password. La password è necessaria per modificare la CPU (scrivervi) e cambiarne il modo di funzionamento (RUN/STOP).
Protetto in lettura ed in scrittura	Consente l'accesso HMI e tutti i tipi di comunicazione da PLC a PLC senza password. La password è necessaria per leggere i dati dalla CPU, modificarla (scrivervi) e cambiarne il modo di funzionamento (RUN/STOP).

3.1.5 Ripristino in caso di perdita della password

Se si perde la password con cui si è protetta una CPU si deve cancellare il programma protetto con una scheda di trasferimento vuota. La scheda vuota cancella la memoria di caricamento interna della CPU. In seguito sarà possibile caricare un nuovo programma utente da STEP 7 Basic nella CPU.

Per informazioni su come creare e utilizzare le schede di trasferimento vuote consultare il paragrafo sulle schede di trasferimento (Pagina 71).



AVVERTENZA

Se si inserisce una scheda di trasferimento in una CPU in RUN, questa passa in STOP. In condizioni non sicure i dispositivi di comando possono funzionare in modo errato e determinare un funzionamento scorretto delle apparecchiature comandate. Ciò può causare la morte o gravi lesioni al personale e/o danni alle apparecchiature.

Prima di portare la CPU in RUN estrarre la scheda di trasferimento.

3.2 Memorizzazione dei dati, aree di memoria e indirizzamento

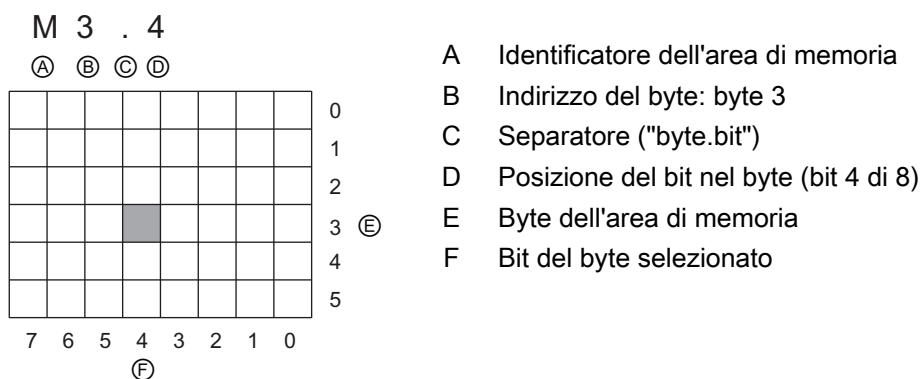
La CPU dispone di svariate opzioni per la memorizzazione dei dati durante l'esecuzione del programma utente:

- **Memoria globale:** La CPU mette a disposizione diverse aree di memoria specializzate, tra cui gli ingressi (I), le uscite (Q) e i merker (M). La memoria è accessibile da tutti i blocchi di codice senza alcuna limitazione.
- **Blocco dati (DB):** nel programma utente si possono inserire dei DB in cui salvare i dati per i blocchi di codice. Quando termina l'esecuzione del blocco di codice associato i dati vengono mantenuti in memoria. I DB "globali" memorizzano dati che possono essere utilizzati da tutti i blocchi di codice mentre i DB di istanza memorizzano solo quelli per un FB specifico e sono strutturati dai parametri dell'FB.
- **Memoria temporanea:** quando viene richiamato un blocco di codice il sistema operativo della CPU assegna la memoria (L) temporanea o locale che potrà essere utilizzata durante l'esecuzione del blocco. Al termine dell'esecuzione la CPU riassegna la memoria locale per l'esecuzione di altri blocchi di codice.

Ogni diversa locazione di memoria ha un indirizzo univoco. Il programma utente si serve di questi indirizzi per accedere alle informazioni contenute nella rispettiva locazione di memoria.

Area di memoria	Descrizione	Forzamento	Ritenzione
I Immagine di processo degli ingressi	Viene copiata dagli ingressi fisici all'inizio del ciclo di scansione	No	No
I_:P (ingresso fisico)	Lettura diretta degli ingressi fisici della CPU e degli SB ed SM	Sì	No
Q Immagine di processo delle uscite	Copiata nelle uscite fisiche all'inizio del ciclo di scansione	No	No
Q_:P (uscita fisica)	Scrittura diretta nelle uscite fisiche della CPU e degli SB ed SM	Sì	No
M Memoria di merker	Memoria di comando e di dati	No	Sì
L Memoria temporanea	Dati temporanei di un blocco, locali per il blocco	No	No
DB Blocco dati	Memoria di dati e, nel caso degli FB, anche memoria per i parametri	No	Sì

Ogni diversa locazione di memoria ha un indirizzo univoco. Il programma utente si serve di questi indirizzi per accedere alle informazioni contenute nella rispettiva locazione di memoria. La figura mostra le modalità di accesso a un bit (detto anche indirizzamento "byte.bit"). In questo esempio l'area di memoria e l'indirizzo del byte (I = ingresso e 3 = byte 3) sono seguiti da un punto decimale (".") che separa l'indirizzo del bit (bit 4).



Il formato "Indirizzo a byte" consente di accedere alla maggior parte delle aree di memoria (I, Q, M, DB ed L) come byte, parole o doppie parole. Per accedere a un byte, una parola o una doppia parola di dati della memoria se ne deve specificare l'indirizzo in modo analogo all'indirizzo di un bit. L'indirizzo è costituito dall'identificatore di area, dalla grandezza dei dati e dall'indirizzo byte iniziale del byte, della parola o della doppia parola. B (byte), W (word - parola) e D (double-word - doppia parola) indicano le dimensioni (ad es. IB0, MW20 o QD8). I riferimenti quali I0.3 e Q1.7 accedono all'immagine di processo. Per accedere a un ingresso o un'uscita fisica aggiungere ":P" al riferimento (ad es. I0.3:P, Q1.7:P o "Stop:P").

Accesso ai dati delle aree di memoria della CPU

STEP 7 Basic facilita la programmazione simbolica. Generalmente le variabili vengono create nelle variabili PLC, in un blocco dati o nell'interfaccia sopra un OB, un FC o un FB. Le variabili sono costituite dal nome, dal tipo di dati, dall'offset e dal commento. Nei blocchi dati si può specificare anche un valore iniziale. Per utilizzare queste variabili mentre si programma se ne deve specificare il nome nel parametro dell'istruzione. In opzione si può indicare l'operando assoluto (memoria, area, dimensione e offset) nel parametro dell'istruzione. Gli esempi riportati nei prossimi capitoli spiegano come immettere gli operandi assoluti. L'editor di programma inserisce automaticamente il carattere % prima dell'operando assoluto. Il tipo di visualizzazione dell'editor di programma può essere scelto tra uno dei seguenti: simbolica, simbolica e assoluta o assoluta.

I (immagine di processo degli ingressi): la CPU campiona gli ingressi (fisici) della periferia immediatamente prima dell'esecuzione dell'OB di ciclo di ogni ciclo di scansione e scrive i valori rilevati nell'immagine di processo degli ingressi. L'accesso all'immagine di processo degli ingressi può essere effettuato a bit, byte, parola e doppia parola. È consentito l'accesso sia in scrittura che in lettura, ma generalmente gli ingressi dell'immagine di processo vengono solo letti.

Bit	I[indirizzo byte].[indirizzo bit]	I0.1
Byte, parola o doppia parola	I[dimensione][indirizzo byte iniziale]	IB4, IW5 o ID12

Aggiungendo ":P" all'indirizzo si fa in modo che gli ingressi digitali e analogici della CPU, dell'SB o dell'SM vengano letti immediatamente. La differenza tra un accesso mediante I_:P invece che I consiste nel fatto che i dati provengono direttamente dall'ingresso a cui si accede invece che dall'immagine di processo degli ingressi. L'accesso I_:P è considerato una "lettura diretta" perché i dati vengono prelevati direttamente dall'origine invece che dalla copia dell'ultima immagine di processo degli ingressi aggiornata.

Poiché gli ingressi fisici ricevono i loro valori direttamente dall'apparecchiatura da campo a cui sono collegati, non è consentito scrivervi. Ciò significa che gli accessi I_:P sono di sola lettura, diversamente dagli accessi I che possono essere di lettura o di scrittura.

Gli accessi I_:P sono inoltre limitati alla dimensione degli ingressi supportati da una singola CPU o modulo SB o SM, arrotondata al byte più vicino. Se, ad esempio, gli ingressi 2 DI / 2 DQ di un SB sono stati configurati in modo da iniziare da I4.0, vi si può accedere con I4.0:P e I4.1:P oppure IB4:P. Gli accessi a I4.2:P - I4.7:P vengono comunque accettati ma non hanno senso perché non vengono utilizzati. Gli accessi a IW4:P e ID4:P non sono consentiti perché superano l'offset di byte associato all'SB.

Gli accessi con I_:P non influiscono sul valore corrispondente memorizzato nell'immagine di processo degli ingressi.

Bit	I[indirizzo byte].[indirizzo bit]:P	I0.1:P
Byte, parola o doppia parola	I[dimensione][indirizzo byte iniziale]:P	IB4:P, IW5:P o ID12:P

Q (immagine di processo delle uscite): La CPU copia nelle uscite fisiche i valori memorizzati nell'immagine di processo delle uscite. L'accesso all'immagine di processo delle uscite può essere effettuato a bit, byte, parola e doppia parola. È consentito l'accesso sia in lettura che in scrittura.

Bit	Q[indirizzo byte].[indirizzo bit]	Q1.1
Byte, parola o doppia parola	Q[dimensione][indirizzo byte iniziale]	QB5, QW10, QD40

Aggiungendo ":P" all'indirizzo si fa in modo che le uscite fisiche digitali e analogiche della CPU, dell'SB o dell'SM vengano scritte immediatamente. La differenza tra un accesso mediante Q_:P invece di Q consiste nel fatto che i dati vengono scritti direttamente nell'uscita a cui si accede oltre che nell'immagine di processo delle uscite (la scrittura viene effettuata in entrambi i punti). L'accesso Q_:P a volte viene definito "scrittura diretta" perché i dati vengono inviati direttamente all'uscita di destinazione, che non deve quindi attendere il successivo aggiornamento dell'immagine di processo delle uscite.

Poiché le uscite fisiche comandano direttamente le apparecchiature da campo a cui sono collegate non è consentito leggerle. Ciò significa che gli accessi Q_:P sono di sola scrittura, diversamente dagli accessi Q che possono essere di lettura o di scrittura.

Gli accessi Q_:P sono inoltre limitati alla dimensione delle uscite supportate da una singola CPU o modulo SB o SM, arrotondata al byte più vicino. Se, ad esempio, le uscite 2 DI / 2 DQ di un SB sono state configurate in modo da iniziare da Q4.0, vi si può accedere con Q4.0:P e Q4.1:P oppure QB4:P. Gli accessi a Q4.2:P - Q4.7:P vengono comunque accettati ma non hanno senso perché non vengono utilizzati. Gli accessi a QW4:P e QD4:P non sono consentiti perché superano l'offset di byte associato all'SB.

Gli accessi con Q_:P influiscono sia sull'uscita fisica che sul corrispondente valore memorizzato nell'immagine di processo delle uscite.

Bit	Q[indirizzo byte].[indirizzo bit]:P	Q1.1:P
Byte, parola o doppia parola	Q[dimensione][indirizzo byte iniziale]:P	QB5:P, QW10:P o QD40:P

M (area dei merker): l'area dei merker (memoria M) può essere utilizzata sia per i relè di controllo che per i dati al fine di memorizzare lo stato intermedio di un'operazione o altre informazioni di comando. L'accesso all'area dei merker può essere effettuato a bit, byte, parola e doppia parola. È consentito l'accesso sia in lettura che in scrittura.

Bit	M[indirizzo byte].[indirizzo bit]	M26.7
Byte, parola o doppia parola	M[dimensione][indirizzo byte iniziale]	MB20, MW30, MD50

Temp (memoria temporanea): La CPU assegna la memoria temporanea in base alle necessità. Quella per il blocco di codice viene assegnata all'avvio (nel caso degli OB) o al richiamo (nel caso delle FC o degli FB) del blocco. È possibile che vengano assegnati a un blocco di codice gli stessi indirizzi di memoria temporanea precedentemente utilizzati da un'altra FC o un altro OB o FB. La CPU non inizializza la memoria temporanea durante l'assegnazione, per cui la memoria può contenere qualsiasi valore.

La memoria temporanea è simile alla memoria M con un'eccezione fondamentale: la memoria M è "globale" mentre la memoria L è "locale":

- Memoria M: qualsiasi OB, FC o FB può accedere ai dati di questa area di memoria, ovvero i dati sono disponibili globalmente per tutti gli elementi del programma utente.
- Memoria temporanea: l'accesso ai dati di questa area è limitato all'OB, l'FC o l'FB che ha creato o dichiarato l'indirizzo di memoria temporanea. Gli indirizzi restano locali e non sono condivisi da blocchi di codice diversi, neppure se il blocco di codice ne richiama un altro. Ad esempio: quando un OB richiama un'FC, quest'ultima non può accedere alla memoria temporanea dell'OB da cui è stata richiamata.

La CPU mette a disposizione una memoria temporanea (locale) per ciascuna delle tre classi di priorità degli OB:

- 16 Kbyte per gli OB di avvio e di ciclo compresi gli FB e le FC associati
- 4 Kbyte per gli eventi di allarme standard compresi gli FB e le FC
- 4 Kbyte per gli eventi di allarme di errore compresi gli FB e le FC

L'accesso alla memoria temporanea può essere effettuato esclusivamente tramite indirizzamento simbolico.

DB (blocco dati): i DB possono essere utilizzati per memorizzare diversi tipi di dati, tra cui lo stato intermedio di un'operazione o altri parametri delle informazioni di comando per gli FB e strutture di dati per varie istruzioni, quali i temporizzatori e i contatori. È possibile specificare che il blocco dati sia di lettura/scrittura o di sola lettura. L'accesso alla memoria dei blocchi dati può essere effettuato a bit, byte, parola e doppia parola. Per i blocchi dati di lettura/scrittura è consentito l'accesso sia in lettura che in scrittura. Per i blocchi di sola lettura è consentito solo l'accesso in lettura.

Bit	DB[numero blocco dati].DBX[indirizzo byte].[indirizzo bit]	DB1.DBX2.3
Byte, parola o doppia parola	DB[numero blocco dati].DB [dimensione][indirizzo byte iniziale]	DB1.DBB4, DB10.DBW2, DB20.DBD8

Indirizzamento degli I/O nella CPU e nei moduli di I/O



Vista generale dispositivi						
Unità	Posto	Indirizzo I	Indirizzo Q	Tipo	N° di o	
RS485_1	103			CM 1241 (RS485)	6ES7	
	102					
PLC_1	101			CPU 1214C DC/DC	6ES7	
DI14/DO10	1.1	0...1	0...1	DI14/DO10		
AI2	1.2	64...67		AI2		
AO1 x 12bit	1.3		80...81	Signal board AO1	6ES7	
HSC_1	1.16	1000...		Contatori veloci (H)		
HSC_2	1.17			Contatori veloci (H)		
HSC_3	1.18			Contatori veloci (H)		
HSC_4	1.19			Contatori veloci (H)		
HSC_5	1.20			Contatori veloci (H)		
HSC_6	1.21			Contatori veloci (H)		
Pulse_1	1.32			Generatore di imp.		
Pulse_2	1.33			Generatore di imp.		
Interfaccia ... X1				Interfaccia PROFIBUS		
DIB x DC24V..	2	8		SM 1221 DIB x DC	6ES7	

Quando si aggiungono una CPU e dei moduli di I/O nella schermata di configurazione, vengono assegnati automaticamente gli indirizzi I e Q.

L'indirizzamento di default può essere modificato selezionando il campo dell'indirizzo nella schermata di configurazione e digitando nuovi numeri. Gli ingressi e le uscite digitali vengono assegnati in byte completi di 8 bit, a prescindere dal fatto che il modulo li utilizzi tutti o meno. Gli ingressi e le uscite analogiche vengono assegnati in gruppi di 2 (4 byte). Nell'esempio qui raffigurato si potrebbe modificare l'indirizzo di DI16 in 2..3 invece che 8..9. Il software facilita l'operazione modificando i campi degli indirizzi che hanno una dimensione errata o che entrano in conflitto con altri indirizzi.

La seguente figura mostra un esempio di CPU 1214C con due SM.

3.3 Tipi di dati

I tipi di dati consentono sia di specificare la dimensione di un elemento di dati che di indicare come vanno interpretati i dati. Ciascun parametro di un'istruzione supporta almeno un tipo di dati e alcuni ne supportano più di uno. Posizionando il cursore sul campo del parametro di un'istruzione si possono visualizzare i tipi di dati supportati.

Il parametro formale di un'istruzione è l'identificatore che indica la posizione dei dati che questa deve utilizzare (ad esempio: l'ingresso IN1 di un'istruzione ADD). Il parametro attuale corrisponde all'indirizzo di memoria o alla costante in cui si trovano i dati che devono essere utilizzati dall'istruzione (ad esempio %MD400 "Numero_di_oggetti"). Il tipo di dati del parametro attuale specificato dall'utente deve essere supportato dal parametro formale specificato dall'istruzione.

Quando si specifica un parametro attuale si deve indicare una variabile (simbolo) o un indirizzo di memoria assoluto. Le variabili associano un nome simbolico (il nome della variabile) a un tipo di dati, un'area di memoria, un offset di memoria e un commento, e possono essere create nell'editor delle variabili PLC o nell'editor di interfaccia di un blocco (OB, FC, FB o DB). Se si indica un indirizzo assoluto a cui non è stata associata una variabile si deve utilizzare una dimensione appropriata al tipo di dati supportato, in modo che, una volta immesso l'indirizzo, venga creata automaticamente una variabile di default.

3.3 Tipi di dati

Per molti parametri di ingresso si può indicare anche un valore costante. La seguente tabella descrive i tipi di dati semplici supportati e alcuni esempi di costanti. Tutti, tranne String, sono disponibili nell'editor delle variabili PLC e negli editor di interfaccia di blocco. String è disponibile solo negli editor di interfaccia di blocco. La seguente tabella definisce i tipi di dati semplici.

Tipo di dati	Dimensione (bit)	Campo	Esempi di costanti
Bool	1	Da 0 a 1	TRUE, FALSE, 0, 1
Byte	8	Da 16#00 a 16#FF	16#12, 16#AB
Word	16	Da 16#0000 a 16#FFFF	16#ABCD, 16#0001
DWord	32	Da 16#00000000 a 16#FFFFFFFF	16#02468ACE
Char	8	Da 16#00 a 16#FF	'A', 't', '@'
Sint	8	Da -128 a 127	123, -123
Int	16	Da -32.768 a 32.767	123, -123
Dint	32	Da -2.147.483.648 a 2.147.483.647	123, -123
USInt	8	Da 0 a 255	123
UInt	16	Da 0 a 65.535	123
UDInt	32	Da 0 a 4.294.967.295	123
Real	32	Da +/-1,18 x 10 ⁻³⁸ a +/-3,40 x 10 ³⁸	123.456, -3.4, -1.2E+12, 3.4E-3
LReal	64	Da +/-2.23 x 10 ⁻³⁰⁸ a +/-1.79 x 10 ³⁰⁸	12345.123456789 -1.2E+40
Time	32	T#-24d_20h_31m_23s_648ms ... T#24d_20h_31m_23s_647ms Salvati come: -2,147,483,648 ms ... +2,147,483,647 ms	T#5m_30s 5#-2d T#1d_2h_15m_30x_45ms
String	Variabile	Da 0 a 254 caratteri in formato di byte	'ABC'

Il seguente formato numerico BCD è supportato dalle istruzioni di conversione nonostante non sia disponibile come tipo di dati.

Formato	Dimensione (bit)	Campo numerico	Esempi di costanti
BCD16	16	Da -999 a 999	123, -123
BCD32	32	Da -9999999 a 9999999	1234567, -1234567

Formato dei numeri reali

I numeri reali (o in virgola mobile) sono costituiti dai numeri a 32 bit a precisione singola (Real) o dai numeri a 64 bit a precisione doppia (LReal) aventi il formato descritto nello standard ANSI/IEEE754-1985. I numeri in virgola mobile a precisione singola sono precisi fino alla sesta cifra significativa e quelli a precisione doppia fino alla quindicesima. Per mantenere la precisione, quando si specificano le costanti in virgola mobile si possono indicare al massimo 6 (Real) o 15 cifre significative (LReal).

I calcoli che includono una lunga serie di valori con numeri molto grandi o molto piccoli possono dare risultati imprecisi. Ciò accade se i numeri differiscono di 10 elevato alla potenza di x, dove x > 6 (Real) o 15 (LReal).

Ad esempio (Real): 100 000 000 + 1 = 100 000 000.

Formato del tipo di dati String

La CPU supporta il tipo di dati STRING che consente di memorizzare una sequenza di caratteri di un byte. Questo tipo di dati contiene il numero massimo di caratteri (della stringa) e il numero di caratteri attuali. STRING mette a disposizione 256 byte per memorizzare il numero massimo di caratteri (1 byte), il numero di caratteri attuali (1 byte) e fino a un massimo di 254 caratteri, ognuno memorizzato in 1 byte.

Se si usano le virgolette singole è possibile utilizzare letterali di stringa (costanti) per i parametri delle istruzioni di tipo IN. Ad esempio 'ABC' è una stringa di tre caratteri che può essere utilizzata come ingresso per il parametro IN dell'istruzione S_CONV. È inoltre possibile creare variabili di stringa selezionando il tipo di dati "String" negli editor di interfaccia dei blocchi OB, FC, FB e DB. Non è possibile creare stringhe nell'editor delle variabili PLC. Quando si dichiara una stringa se ne può specificare la dimensione massima; ad esempio "MyString[10]" indica che MyString può avere una dimensione massima di 10 byte. Se non si indicano le parentesi quadre con le dimensioni massime, queste vengono impostate automaticamente a 254.

Il seguente esempio definisce una stringa con un numero massimo di caratteri pari a 10 e un numero di caratteri attuali pari a 3. La stringa contiene quindi 3 caratteri di un byte, ma potrebbe essere ingrandita fino a 10 caratteri di 1 byte.

Numero massimo di caratteri	Numero di caratteri attuali	Carattere 1	Carattere 2	Carattere 3	...	Carattere 10
10	3	'C' (16#43)	'A' (16#41)	'T' (16#54)	...	-
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	...	Byte 11

Array

È possibile creare un array che contiene più elementi con tipo di dati semplice. Per creare gli array si utilizzano gli editor di interfaccia dei blocchi OB, FC, FB e DB. Non è possibile creare un array nell'editor delle variabili PLC.

Per creare un array nell'editor di interfaccia di un blocco, selezionare il tipo di dati "Array [lo .. hi] of type", quindi indicare "lo", "hi" e "type" nel seguente modo:

- lo - l'indice iniziale (più basso) dell'array
- hi - l'indice finale (più alto) dell'array
- type - un tipo di dati semplici, ad es. BOOL, SINT, UDINT

Vengono supportati gli indici negativi. Nella colonna Nome dell'editor di interfaccia del blocco è possibile attribuire un nome all'array. La seguente tabella mostra alcuni esempi di array così come verrebbero visualizzati nell'editor di interfaccia del blocco:

Nome	Tipo di dati	Commento
My_Bits	Array [1 .. 10] of BOOL	Questo array contiene 10 valori booleani
My_Data	Array [-5 .. 5] of SINT	Questo array contiene 11 valori SINT, compreso l'indice 0

Per indirizzare gli elementi degli array nel programma si utilizza la seguente sintassi:

- Array_name[i], dove i è l'indice desiderato.

Esempi di array in ingresso a un parametro, così come verrebbero visualizzati nell'editor di programma:

- #My_Bits[3] - indirizza il terzo bit dell'array "My_Bits"
- #My_Data[-2] - indirizza il quarto SINT dell'array "My_Data"

Il carattere # viene inserito automaticamente dall'editor di programma.

Tipo di dati DTL (Data and Time Long)

Il tipo di dati DTL è una struttura di 12 byte che salva le informazioni sulla data e l'ora in una struttura predefinita. Un DTL può essere definito sia nella memoria temporanea del blocco che in un DB.

Lunghezza (byte)	Formato	Campo di valori	Esempio di valore immesso
12	Orologio e calendario (Anno-Mese variabile:Ora:Minuto:Second o.Nanosecondi)	Min.: DTL#1970-01-01- 00:00:00.0 Max.: DTL#2554-12-31- 23:59:59.999 999 999	DTL#2008-12-16- 20:30:20.250

Ogni componente del DTL contiene un diverso tipo di dati e campo di valori. Il tipo di dati di un valore specificato deve essere uguale a quello dei relativi componenti.

BYTE	Componente	Tipo di dati	Campo di valori
0	Anno	UINT	Da 1970 a 2554
1			
2	Mese	USINT	Da 1 a 12
3	Giorno	USINT	Da 1 a 31
4	Giorno della settimana	USINT	Da 1 (domenica) a 7 (sabato) Il giorno della settimana non viene considerato.
5	Ora	USINT	Da 0 a 23
6	Minuto	USINT	Da 0 a 59
7	Secondo	USINT	Da 0 a 59
8	Nanosecondi	UDINT	Da 0 a 999 999 999
9			
10			
11			

3.4 Utilizzo della memory card

ATTENZIONE

La CPU supporta solo memory card SIMATIC (Pagina 336) preformattate. Se si riformatta la memory card SIMATIC con Windows la CPU non la può utilizzare.

Prima di copiare un programma nella memory card formattata, cancellare quelli eventualmente già presenti.

La memory card può essere utilizzata come scheda di trasferimento o di programma. I programmi che vi vengono copiati contengono tutti i blocchi di codice e di dati, gli oggetti tecnologici e la configurazione dei dispositivi. Il programma **non** contiene i valori forzati.

- La scheda di trasferimento consente di copiare un programma nella memoria di caricamento interna della CPU senza utilizzare STEP 7 Basic. Dopo aver inserito una scheda di trasferimento la CPU cancella il programma utente e gli eventuali valori forzati dalla memoria di caricamento interna e vi copia il programma della scheda. Al termine del trasferimento si deve estrarre la scheda.

Se si perde o si dimentica la password (Pagina 60) che protegge una CPU, per accedervi si può utilizzare una scheda di trasferimento vuota. Quando la si inserisce il programma protetto viene cancellato dalla memoria di caricamento interna della CPU. Quindi si può procedere al caricamento di un nuovo programma.

- La scheda di programma può essere utilizzata come memoria di caricamento esterna della CPU. Quando la si inserisce, la memoria di caricamento interna della CPU viene interamente cancellata (il programma utente e gli eventuali valori forzati). La CPU esegue quindi il programma nella memoria di caricamento esterna (la scheda). Se si effettua il caricamento in una CPU in cui è stata inserita una scheda di programma, viene aggiornata solo la memoria di caricamento esterna (cioè la scheda).

Poiché la memoria di caricamento interna della CPU è stata cancellata quando è stata inserita la scheda di programma, questa **deve** rimanere nella CPU. Se la si estrae la CPU passa in STOP (il LED di errore lampeggia per indicare che la scheda è stata estratta).

Il programma della memory card comprende i blocchi di codice, i blocchi dati, gli oggetti tecnologici e la configurazione dei dispositivi. **Non** contiene invece i valori forzati. Questi non fanno parte del programma e vengono memorizzati nella memoria di caricamento oppure nella memoria interna o esterna della CPU (la scheda di programma). Se si inserisce una scheda di programma nella CPU, STEP 7 Basic applica i valori forzati solo alla memoria di caricamento esterna (la scheda).

3.4.1 Inserimento di una memory card nella CPU

AVVERTENZA

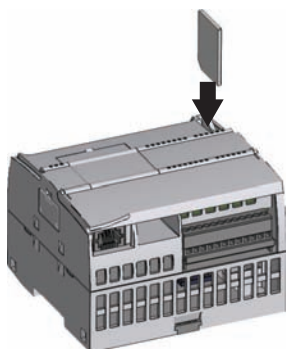
Se si inserisce una memory card (configurata come scheda di programma o di trasferimento) in una CPU in RUN, la CPU passa immediatamente in STOP. In condizioni non sicure i dispositivi di comando possono funzionare in modo errato e determinare un funzionamento scorretto delle apparecchiature comandate. Ciò può causare la morte o gravi lesioni al personale e/o danni alle apparecchiature. Installare sempre un circuito di arresto d'emergenza per l'applicazione o il processo.

CAUTELA

Le scariche elettrostatiche possono danneggiare la memory card o il relativo alloggiamento nella CPU.

Quando la si maneggia, si deve toccare una superficie metallica messa a terra e/o indossare una fascetta di messa a terra. È inoltre necessario custodire la memory card in un contenitore a conduzione di corrente.

Per inserire una memory card aprire il coperchio sopra la CPU e inserire la scheda nello slot. L'inserimento e l'estrazione sono facilitate dal connettore di tipo push-push. La memory card è realizzata in modo tale da impedire un inserimento errato.



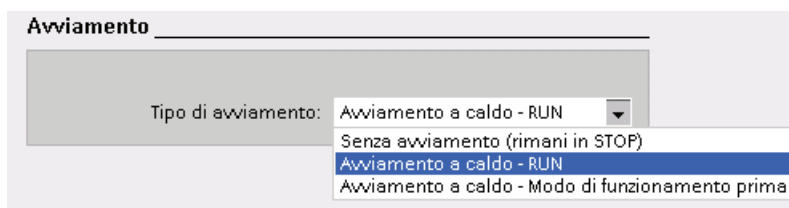
Verificare che la memory card non sia protetta dalla scrittura. Sbloccare la levetta di protezione.

Nota

Se si inserisce una memory card con la CPU in STOP, il buffer di diagnostica visualizza un messaggio indicante l'inizializzazione della valutazione della memory card. Non tenere conto di questo messaggio. La valutazione della memory card non inizia fino a quando non si commuta la CPU in RUN, si resetta la memoria della CPU con una cancellazione totale (MRES) o si riaccende la CPU.

3.4.2 Configurazione del parametro di avvio della CPU prima di copiare il progetto nella memory card

Il programma copiato in una scheda di trasferimento o di programma contiene il parametro di avvio per la CPU. Prima di copiare il programma nella memory card verificare di aver configurato il modo di funzionamento da impostare in seguito allo spegnimento/riaccensione della CPU. Selezionare se la CPU si avvierà in STOP, in RUN o con il modo di funzionamento precedente allo spegnimento/riaccensione.



3.4.3 Scheda di trasferimento

CAUTELA

Le scariche elettrostatiche possono danneggiare la memory card o il relativo alloggiamento nella CPU.

Quando la si maneggia, si deve toccare una superficie metallica messa a terra e/o indossare una fascetta di messa a terra. È inoltre necessario custodire la memory card in un contenitore a conduzione di corrente.

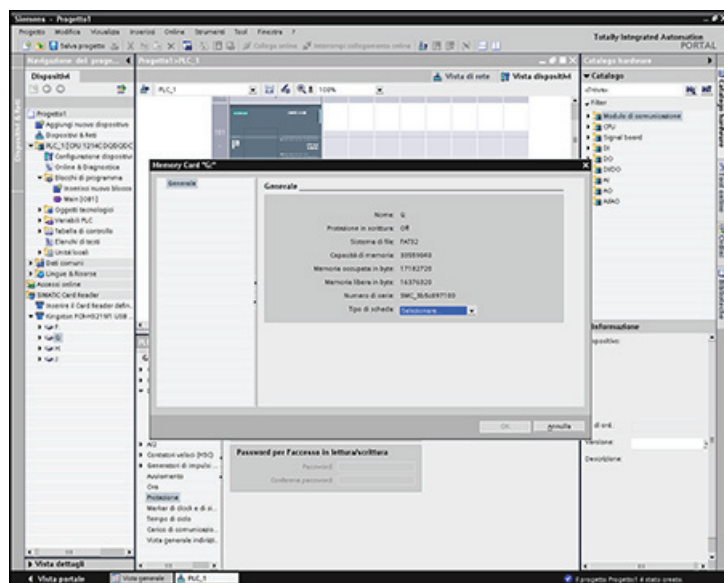
Creazione di una scheda di trasferimento

Prima di copiare un programma nella scheda di trasferimento ricordarsi sempre di configurare il parametro di avvio della CPU (Pagina 71). Per creare una scheda di trasferimento procedere nel seguente modo:

1. Inserire una memory card vuota nel dispositivo di lettura/scrittura schede collegato al dispositivo di programmazione
(se la memory card non è vuota cancellare la cartella "SIMATIC.S7S" e il file "S7_JOB.S7S" utilizzando un'applicazione come Esplora risorse di Windows).
2. Nell'albero del progetto (vista progetto), espandere la cartella "SIMATIC Card Reader" e selezionare il lettore di schede.
3. Visualizzare la finestra di dialogo "Memory card" facendo clic con il tasto destro del mouse sulla memory card nel lettore di schede e selezionando "Proprietà" nel menu di riepilogo.

4. Selezionare "Trasmissione" nel menu a discesa della finestra di dialogo "Memory card".

A questo punto STEP 7 Basic crea la scheda di trasferimento vuota. Se si sta creando una scheda di trasferimento vuota, come per il ripristino in caso di perdita della password della CPU (Pagina 60), estrarre la scheda di trasferimento dal lettore di schede.



5. Aggiungere il programma selezionando la CPU (ad es. PLC_1 [CPU 1214 DC/DC/DC]) nell'albero del progetto e trascinandola nella memory card (un altro metodo consiste nel copiare la CPU e incollarla nella memory card). Quando si copia la CPU nella memory card si apre la finestra di dialogo "Carica anteprima".
6. Fare clic sul pulsante "Carica" della finestra per copiare la CPU nella memory card.
7. Quando la finestra di dialogo visualizza un messaggio indicante che il programma della CPU è stato caricato senza errori, fare clic sul pulsante "Fine".

Utilizzo di una scheda di trasferimento

Per trasferire il programma in una CPU procedere nel seguente modo:

1. Inserire la scheda di trasferimento nella CPU (Pagina 70). Se la CPU è in RUN, commuta in STOP (il LED di manutenzione lampeggia per segnalare che è necessario valutare la memory card).
2. Per procedere utilizzare una delle seguenti opzioni:
 - Spegner e riaccendere la CPU.
 - Eseguire una commutazione da STOP a RUN.
 - Eseguire un reset della memoria (MRES).
3. Dopo il riavvio e la valutazione della memory card, la CPU copia il programma nella propria memoria di caricamento interna. Al termine dell'operazione di copia il LED di manutenzione della CPU lampeggia per segnalare che la scheda di trasferimento può essere estratta.
4. Estrarre dalla CPU la memory card di "trasferimento".
5. Per valutare il nuovo programma trasferito nella memoria di caricamento interna utilizzare una delle seguenti opzioni:
 - Spegner e riaccendere la CPU.
 - Eseguire una commutazione da STOP a RUN.
 - Eseguire un reset della memoria (MRES).

La CPU passa nel modo di avvio (RUN o STOP) configurato per il progetto.

Nota

Prima di portare la CPU in RUN estrarre la scheda di trasferimento.

3.4.4 Scheda di programma

CAUTELA

Le scariche elettrostatiche possono danneggiare la memory card o il relativo alloggiamento nella CPU.

Quando la si maneggia, si deve toccare una superficie metallica messa a terra e/o indossare una fascetta di messa a terra. È inoltre necessario custodire la memory card in un contenitore a conduzione di corrente.



Verificare che la memory card non sia protetta dalla scrittura. Sbloccare la levetta di protezione.

Prima di copiare gli elementi di un programma nella memory card, cancellare i programmi eventualmente già presenti.

Creazione di una scheda di programma

Se utilizzata come scheda di programma, la memory card funge da memoria esterna della CPU. Se si estrae la scheda di programma, la memoria di caricamento interna della CPU è vuota.

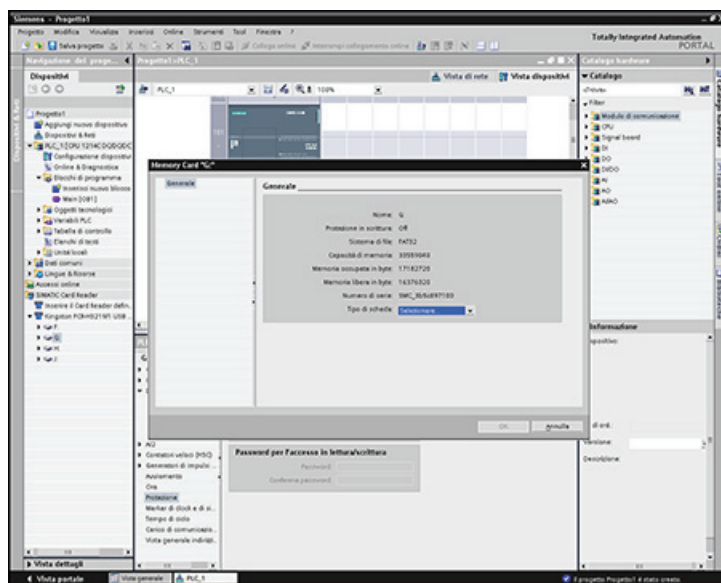
Nota

Se si inserisce una memory card vuota nella CPU e la si valuta spegnendo e riaccendendo la CPU, commutando da STOP a RUN o resettando la memoria (MRES), il programma e i valori forzati presenti nella memoria di caricamento interna della CPU vengono copiati nella memory card (ora la memory card è una scheda di programma). Al termine della copia il programma nella memoria di caricamento interna della CPU viene cancellato. La CPU passa nel modo di avvio configurato (RUN o STOP).

Ricordarsi sempre di configurare il parametro di avvio della CPU (Pagina 71) prima di copiare il progetto nella scheda di programma. Per creare una scheda di programma con STEP 7 Basic procedere nel seguente modo:

1. Inserire una memory card vuota nel dispositivo di lettura/scrittura schede collegato al dispositivo di programmazione
(se la memory card non è vuota cancellare la cartella "SIMATIC.S7S" e il file "S7_JOB.S7S" utilizzando un'applicazione come Esplora risorse di Windows).
2. Nell'albero del progetto (vista progetto), espandere la cartella "SIMATIC Card Reader" e selezionare il lettore di schede.
3. Visualizzare la finestra di dialogo "Memory card" facendo clic con il tasto destro del mouse sulla memory card nel lettore di schede e selezionando "Proprietà" nel menu di riepilogo.

4. Selezionare "Programma" nel menu a discesa della finestra di dialogo "Memory card".



5. Aggiungere il programma selezionando la CPU (ad es. PLC_1 [CPU 1214 DC/DC/DC]) nell'albero del progetto e trascinandola nella memory card (un altro metodo consiste nel copiare la CPU e incollarla nella memory card). Quando si copia la CPU nella memory card si apre la finestra di dialogo "Carica anteprema".
6. Fare clic sul pulsante "Carica" della finestra per copiare la CPU nella memory card.
7. Quando la finestra di dialogo visualizza un messaggio indicante che il programma della CPU è stato caricato senza errori, fare clic sul pulsante "Fine".

Utilizzo di una scheda di programma come memoria di caricamento della CPU

CAUTELA

Se si inserisce una memory card vuota nella CPU, questa passa in STOP. Se si spegne/riaccende la CPU, la si commuta da STOP a RUN o se ne resetta la memoria (MRES), la CPU copia la propria memoria di caricamento interna nella memory card (configurandola così come scheda di programma) e cancella il programma dalla memoria di caricamento interna. Se si estrae la scheda, la memoria di caricamento interna della CPU non contiene alcun programma.

Per utilizzare una scheda di programma nella CPU procedere nel seguente modo:

1. Inserire la scheda di programma nella CPU. Se la CPU è in RUN passa in STOP. Il LED di manutenzione lampeggia per segnalare che è necessario valutare la scheda di programma.
2. Per procedere utilizzare una delle seguenti opzioni:
 - Spegnere e riaccendere la CPU.
 - Eseguire una commutazione da STOP a RUN.
 - Eseguire un reset della memoria (MRES).
3. La CPU si riavvia. Dopo il riavvio e la valutazione della scheda di programma la CPU cancella la propria memoria di caricamento interna.

Quindi passa nel modo di avvio (RUN o STOP) configurato.

La scheda di programma deve rimanere nella CPU. Quando si estrae la scheda di programma, la memoria di caricamento interna della CPU rimane senza programma.



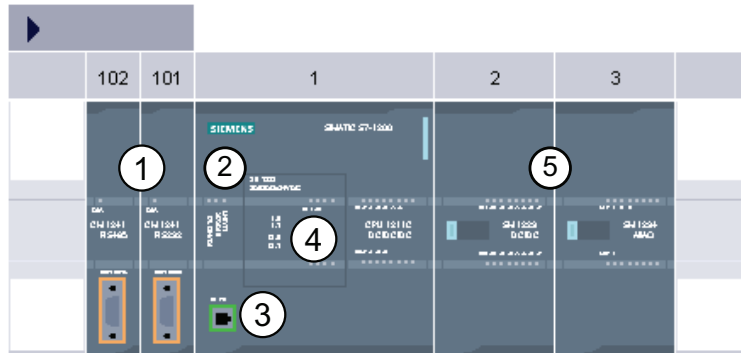
AVVERTENZA

Se si estrae la scheda di programma la CPU perde la memoria di caricamento esterna e genera un errore. La CPU passa in STOP e il LED di errore lampeggia.

In condizioni non sicure i dispositivi di comando possono funzionare in modo errato e determinare un funzionamento scorretto delle apparecchiature comandate. Ciò può causare la morte o gravi lesioni al personale e/o danni alle apparecchiature.

Configurazione dei dispositivi

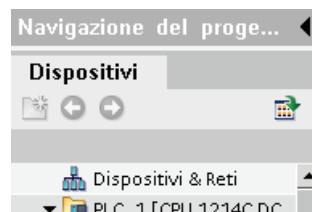
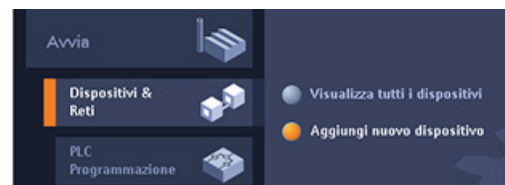
Per creare la configurazione hardware del PLC si devono inserire nel progetto una CPU e altri moduli.



- ① Modulo di comunicazione (CM): fino a 3, inseriti nei posti connettore 101, 102 e 103
- ② CPU: posto connettore 1
- ③ Porta Ethernet della CPU
- ④ Signal board (SB): 1 al massimo, inserita nella CPU
- ⑤ Modulo di I/O (SM) per I/O digitali e analogici: fino a 8, inseriti nei posti connettore da 2 a 9 (la CPU 1214C ne consente 8, la CPU 1212C 2, la CPU 1211C nessuno)

Per creare la configurazione è necessario inserire un dispositivo nel progetto.

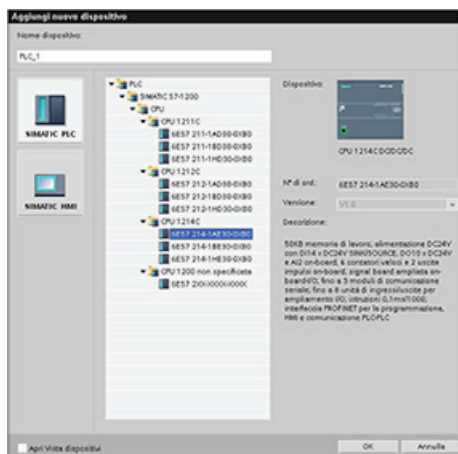
- Selezionare "Dispositivi e reti" nella vista portale e fare clic su "Inserisci dispositivo".
- Nella vista progetto fare doppio clic su "Aggiungi nuovo dispositivo" sotto il nome del progetto.



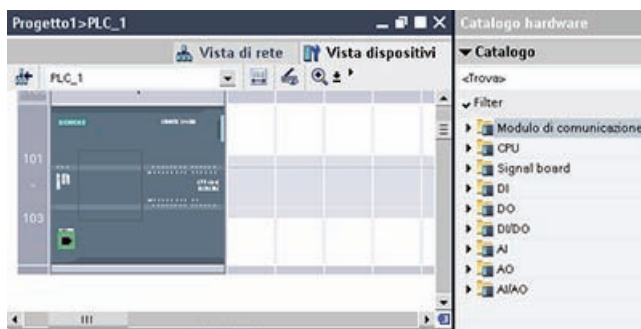
4.1 Inserimento di una CPU

Per creare la configurazione dei dispositivi si deve innanzitutto inserire una CPU nel progetto. Selezionando la CPU nella finestra di dialogo "Aggiungi nuovo dispositivo" vengono creati automaticamente il telaio di montaggio e la CPU.

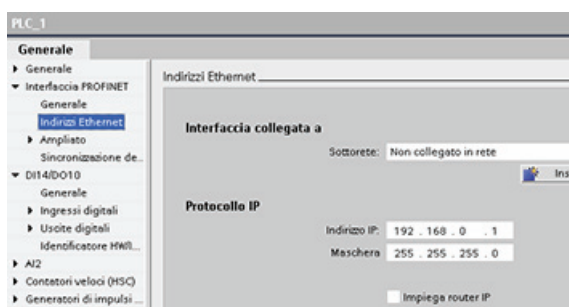
Finestra di dialogo "Aggiungi nuovo dispositivo"



Vista dispositivi della configurazione hardware



Se si seleziona la CPU nella vista dispositivi la finestra di ispezione ne visualizza le proprietà.

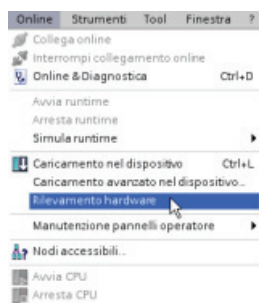


Nota

Poiché la CPU non dispone di un indirizzo IP preconfigurato, l'utente lo deve impostare manualmente durante la configurazione dei dispositivi. Se la CPU è collegata al router di una rete si deve specificare anche l'indirizzo IP del router.

4.2 Rilevamento della configurazione per una CPU non specificata

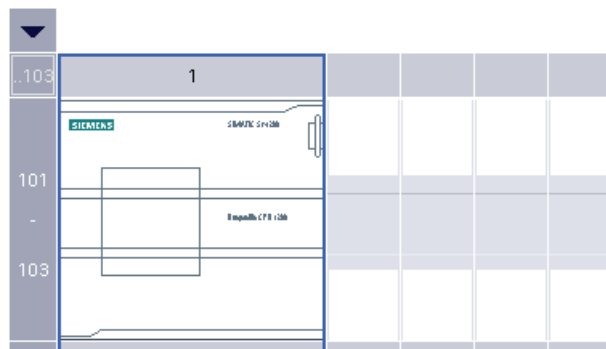
Caricare una configurazione hardware esistente è veramente facile



Se si è collegati a una CPU è possibile caricarne la configurazione, compresi tutti i moduli, nel proprio progetto. Basta creare un nuovo progetto e selezionare la "CPU non specificata" anziché una specifica. (È anche possibile saltare completamente la configurazione dei dispositivi selezionando "Crea programma PLC" dai "Primi passi". STEP 7 Basic crea quindi automaticamente una CPU non specificata.)

Dall'editor di programma selezionare il comando "Rilevamento hardware" nel menu "Online".

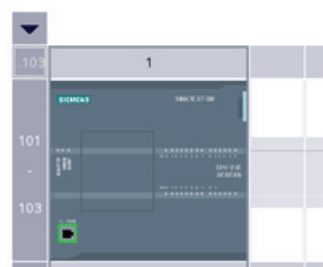
Dall'editor della configurazione dispositivi selezionare l'opzione per il rilevamento della configurazione del dispositivo collegato.



Dispositivo non specificato

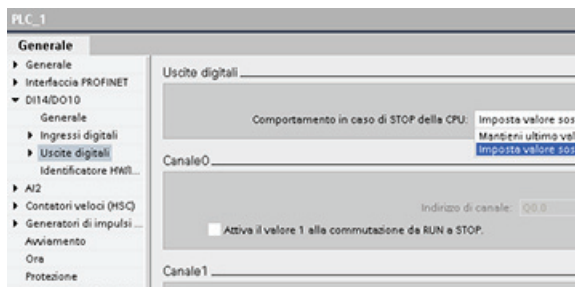
- Utilizzare [Catalogo hardware](#) per specificare la CPU,
- oppure [Rileva](#) la configurazione del dispositivo collegato.

Dopo aver selezionato la CPU dalla finestra di dialogo online, STEP 7 Basic carica la configurazione hardware dalla CPU, compresi tutti i moduli (SM, SB o CM). A questo punto è possibile configurare i parametri per la CPU e i moduli.



4.3 Configurazione del funzionamento della CPU

Per configurare i parametri di esercizio della CPU selezionarla nella vista dispositivi (attorno all'unità compare un riquadro blu) e aprire la scheda "Proprietà" della finestra di ispezione.



Configurare i seguenti parametri nella scheda Proprietà:

- Interfaccia PROFINET: imposta l'indirizzo IP per la sincronizzazione della CPU e dell'ora
- DI, DO e AI: configura il comportamento degli I/O digitali e analogici locali (on-board)
- Contatori veloci e Generatori di impulsi: abilita e configura i contatori veloci (HSC) e i generatori di impulsi per le operazioni PTO (uscita di treni di impulsi) e PWM (modulazione dell'ampiezza degli impulsi).

Quando si configurano le uscite della CPU o della signal board come generatori di impulsi (per l'utilizzo con le istruzioni PWM o di controllo base del movimento), gli indirizzi delle uscite corrispondenti (Q0.0, Q0.1, Q4.0 e Q4.1) vengono cancellati dalla memoria Q e non possono essere utilizzati per altri scopi nel programma utente. Se il programma utente scrive un valore in un'uscita utilizzata come generatore di impulsi, la CPU non scrive quel valore nell'uscita fisica.

- Avviamento: seleziona il comportamento della CPU dopo una transizione off-on, ad esempio facendo in modo che si avvii in STOP o passi in RUN dopo un avviamento a caldo
- Ora: imposta l'ora, il fuso orario e l'ora legale
- Protezione: imposta la protezione in lettura/scrittura e la password per l'accesso alla CPU
- Merker di clock e di sistema: consente di selezionare un byte per le funzioni dei "merker di sistema" (bit "primo ciclo", un bit "sempre on" e un bit "sempre off") e un byte per le funzioni dei "merker di clock" (dove ogni bit si attiva e disattiva a una frequenza predefinita).
- Tempo di ciclo: definisce un tempo di ciclo massimo o un tempo di ciclo minimo fisso
- Carico di comunicazione: assegna la percentuale del tempo della CPU da riservare ai task di comunicazione

4.4 Inserimento di moduli nella configurazione

Per aggiungere moduli alla CPU si utilizza il catalogo hardware. Sono disponibili tre tipi di moduli:

- i moduli di I/O (SM) mettono a disposizione I/O digitali o analogici aggiuntivi. Vengono collegati a destra della CPU.
- Le signal board (SB) aggiungono alla CPU un numero limitato di I/O digitali o analogici. L'SB viene installata sul lato anteriore della CPU.
- I moduli di comunicazione (CM) mettono a disposizione un'ulteriore porta di comunicazione (RS232 o RS485) per la CPU. Vengono collegati a sinistra della CPU.

Per inserire un modulo nella configurazione selezionarlo nel catalogo hardware e fare doppio clic o trascinarlo nel posto connettore selezionato.

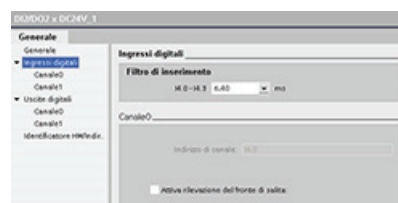
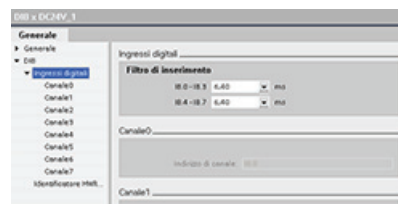
Modulo	Selezione del modulo	Inserimento del modulo	Risultato
SM			
SB			
CM			

4.5 Configurazione dei parametri dei moduli

Per configurare i parametri di esercizio dei moduli selezionare un modulo nella vista dispositivi e aprire la scheda "Proprietà" della finestra di ispezione per configurare i parametri del modulo.

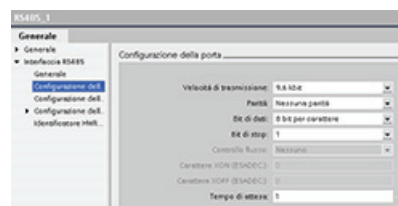
Configurazione di un modulo di I/O (SM) o di una signal board (SB)

- I/O digitali: gli ingressi possono essere configurati per il rilevamento del fronte di salita e di discesa (associandoli ognuno a un evento e un interrupt di processo) e per la "misurazione degli impulsi" (in modo che dopo un impulso istantaneo restino attivi) fino al successivo aggiornamento dell'immagine di processo degli ingressi. Le uscite possono utilizzare un valore congelato o di sostituzione.
- I/O analogici: consente di configurare i parametri dei singoli ingressi, ad esempio il tipo di misura (tensione o corrente), il campo e il livellamento, e di attivare la diagnostica per il controllo dell'underflow o dell'overflow. Le uscite dispongono di parametri per il tipo (tensione o corrente) e la diagnostica, ad es. per i cortocircuiti (nelle uscite in tensione) o i valori limite superiore/inferiore.
- Indirizzi di diagnostica I/O: configura l'indirizzo iniziale del gruppo di ingressi e di uscite del modulo



Configurazione di un modulo di comunicazione (CM)

- Configurazione della porta: configura i parametri di comunicazione, quali velocità di trasmissione, parità, bit di dati, bit di stop, controllo del flusso dati, caratteri XON e XOFF e tempo di attesa
- Configurazione della trasmissione messaggi: attiva e configura le opzioni relative alla trasmissione
- Configurazione della ricezione messaggi: attiva e configura i parametri di inizio e fine dei messaggi



Questi parametri di configurazione possono essere modificati dal programma.

4.6 Creazione di una connessione di rete

Nella "vista di rete" di Configurazione dispositivi si possono creare i collegamenti di rete tra i dispositivi del progetto. Un volta creato un collegamento si possono configurare i parametri della rete nella scheda "Proprietà" della finestra di ispezione.

Azione	Risultato
Selezionare "Vista di rete" per visualizzare i dispositivi da collegare.	
Selezionare la porta di un dispositivo e trascinare il collegamento nella porta del secondo dispositivo.	
Rilasciare il tasto del mouse per creare il collegamento.	

4.7 Configurazione di un indirizzo IP del progetto

Configurazione dell'interfaccia PROFINET

Dopo aver configurato il telaio di montaggio con la CPU (Pagina 80) si possono configurare i parametri per l'interfaccia PROFINET. Fare clic sul quadrato verde PROFINET della CPU e selezionare la porta PROFINET. La scheda "Proprietà" della finestra di ispezione visualizza la porta PROFINET.



① Porta PROFINET

Configurazione dell'indirizzo IP

Indirizzo Ethernet (MAC): in una rete PROFINET ciascun dispositivo è identificato da un indirizzo MAC (acronimo di Media Access Control, ovvero controllo dell'accesso al mezzo fisico) assegnato dal costruttore. Un indirizzo MAC è costituito da sei coppie di cifre esadecimali separate da tratti di congiunzione (-) o da due punti (:) disposte nell'ordine di trasmissione (ad es. 01-23-45-67-89-AB or 01:23:45:67:89:AB).

Indirizzo IP: ogni dispositivo deve avere anche un indirizzo IP (Internet Protocol) che gli consenta di fornire i dati in reti più complesse e provviste di router.

Gli indirizzi IP sono suddivisi in segmenti di 8 bit ed espressi in formato decimale separato da punti (ad esempio: 211.154.184.16). La prima parte dell'indirizzo IP corrisponde all'ID della rete (in quale rete ci si trova?) e la seconda all'ID dell'host (che è unico per ciascun dispositivo della rete). Gli indirizzi IP di tipo 192.168.x.y sono per convenzione indirizzi di reti private che non fanno parte di Internet.

Maschera della sottorete: una sottorete è un raggruppamento logico dei dispositivi collegati ad una rete. Nelle LAN (Local Area Network) i nodi delle sottoreti tendono a essere fisicamente vicini. Le maschere (chiamate "maschere di sottorete" o "maschere di rete") definiscono i limiti delle sottoreti IP.

La maschera di sottorete 255.255.255.0 è generalmente adatta a una rete locale poco estesa. Ciò significa che tutti gli indirizzi IP della rete devono avere gli stessi 3 ottetti iniziali e che i singoli dispositivi sono identificati dall'ultimo ottetto (campo di 8 bit). Ad esempio si possono contrassegnare i dispositivi di una rete locale con la maschera di sottorete 255.255.255.0 e un indirizzo IP da 192.168.2.0 a 192.168.2.255.

Le diverse sottoreti sono collegate solo tramite router. Se si utilizzano le sottoreti è quindi necessario impiegare un router IP.

Router IP: i router costituiscono il collegamento tra le LAN. Tramite il router, il computer di una LAN può trasmettere messaggi a qualsiasi altra rete che può essere a sua volta collegata ad altre LAN. Se la destinazione dei dati non si trova all'interno della LAN, il router li inoltra a un'altra rete o a gruppi di reti da dove possono essere trasmessi alla destinazione.

I router utilizzano gli indirizzi IP per trasmettere e ricevere pacchetti di dati.



Proprietà degli indirizzi IP: nella finestra Proprietà, selezionare il comando di configurazione "Indirizzo Ethernet". Il portale TIA visualizza la finestra di dialogo per la configurazione dell'indirizzo Ethernet, che consente di associare il progetto software all'indirizzo IP della CPU in cui verrà caricato il progetto.

Nota

Poiché la CPU non dispone di un indirizzo IP preconfigurato, lo si deve impostare manualmente. Se la CPU è collegata al router di una rete si deve specificare anche l'indirizzo IP del router. Durante la fase di caricamento vengono configurati tutti gli indirizzi IP.

Per maggiori informazioni consultare il capitolo "Assegnazione degli indirizzi IP ai dispositivi di programmazione e ai dispositivi di rete (Pagina 226)".

La seguente tabella definisce i parametri per l'indirizzo IP:

Parametro		Descrizione
Sottorete		Nome della sottorete a cui è collegato il dispositivo. Per creare una nuova sottorete fare clic sul pulsante "Inserisci nuova sottorete". Per default è impostato "Non collegato in rete". Sono possibili due tipi di collegamento:
		• L'opzione "Non collegato in rete" impostata per default consente di realizzare un collegamento locale. • La sottorete è necessaria se la rete contiene almeno due dispositivi.
Protocollo IP	Indirizzo IP	Indirizzo IP assegnato alla CPU
	Maschera della sottorete	Maschera di sottorete assegnata
	Impiega router IP	Attivare la casella di controllo per indicare che si vuole utilizzare un router IP
	Indirizzo del router	Indirizzo IP assegnato al router (se applicabile)

Concetti di programmazione

5.1 Istruzioni per la progettazione di un sistema PLC

Quando si progetta un sistema PLC si può scegliere tra diversi metodi e criteri. Le seguenti istruzioni generali sono applicabili a svariati progetti. Ovviamente è necessario attenersi alle direttive previste dalle procedure della propria azienda e alle procedure vigenti nel proprio luogo di lavoro e di formazione.

Fasi consigliate	Task
Suddivisione del processo o dell'impianto	Suddividere il processo o l'impianto in parti che siano indipendenti l'una dall'altra. Le parti definiscono i limiti tra i controllori e influiscono sulle specifiche funzionali e l'assegnamento delle risorse.
Creazione delle specifiche funzionali	Descrivere il funzionamento delle singole fasi del processo o dell'impianto, quali gli I/O, la descrizione funzionale delle fasi, gli stati da raggiungere prima di abilitare l'azione degli attuatori (ad es. solenoidi, motori e azionamenti), la descrizione dell'interfaccia operatore e delle eventuali interfacce con altre parti del processo o dell'impianto.
Progettazione dei circuiti di sicurezza	Identificare le apparecchiature che richiedono un cablaggio permanente per motivi di sicurezza. Considerare che i dispositivi di comando possono guastarsi e compromettere la sicurezza del sistema, determinando l'avviamento improvviso o una variazione imprevista del funzionamento delle macchine. Nei casi in cui il funzionamento imprevisto o scorretto delle macchine potrebbe causare lesioni alle persone o gravi danni alle cose, è necessario prevedere dei dispositivi elettromeccanici di esclusione (che intervengano indipendentemente dal PLC) al fine di impedire funzionamenti pericolosi. Nella progettazione dei circuiti di sicurezza è necessario includere quanto indicato di seguito. • Identificare il funzionamento scorretto o imprevisto degli attuatori che potrebbe risultare pericoloso. • Individuare le condizioni che garantiscono un funzionamento sicuro e indicare come rilevarle indipendentemente dal PLC. • Cercare di prevedere come il PLC influirà sul processo in seguito all'inserimento e al disinserimento dell'alimentazione e prevedere come e quando verranno rilevati gli errori. Utilizzare queste informazioni solo per progettare il funzionamento in condizioni normali e in previsione di anomalie, ma non far affidamento su questo "best case" per garantire la sicurezza del sistema. • Progettare dei dispositivi di esclusione manuali o elettromeccanici che, in caso di pericolo, interrompano il funzionamento dell'impianto indipendentemente dal PLC. • Fornire al PLC adeguate informazioni sullo stato dei circuiti indipendenti, in modo che sia il programma che le interfacce utente dispongano dei dati necessari. • Identificare le eventuali ulteriori norme e dispositivi di sicurezza che possono garantire un funzionamento sicuro del sistema.
Indicazione delle stazioni operatore	Tenendo conto dei requisiti delle specifiche funzionali, realizzare i seguenti schemi delle stazioni operatore: • Prospetto indicante la posizione delle stazioni operatore rispetto al processo o alla macchina. • Schema meccanico dei dispositivi per la stazione operatore, quali monitor, interruttori e indicatori luminosi. • Schemi elettrici con gli I/O del PLC e dei moduli di I/O.

Fasi consigliate	Task
Realizzazione dei disegni della configurazione	Tenendo conto dei requisiti delle specifiche funzionali, realizzare i disegni con la configurazione dei dispositivi di controllo: • Prospetto indicante la posizione dei PLC rispetto al processo o all'impianto. • Schema meccanico del PLC e dei moduli di I/O, compresi i quadri elettrici e altri dispositivi. • Schema elettrico dei PLC e dei moduli di I/O, compresi i codici del tipo di dispositivo, gli indirizzi per la comunicazione e gli indirizzi di I/O.
Creazione di un elenco dei nomi simbolici	Fare un elenco dei nomi simbolici per gli indirizzi assoluti indicando oltre ai segnali degli I/O fisici, anche gli altri elementi (ad es. i nomi delle variabili) che verranno utilizzati nel programma.

5.2 Strutturazione del programma utente

Quando si scrive il programma utente per un task di automazione si inseriscono le necessarie istruzioni in blocchi di codice:

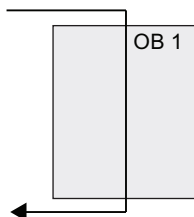
- I blocchi organizzativo (OB) reagiscono a un evento specifico che si verifica nella CPU e possono interrompere l'esecuzione del programma utente. Il blocco di default per l'esecuzione ciclica del programma (OB 1) determina la struttura base del programma utente ed è l'unico blocco di codice effettivamente indispensabile. Se si inseriscono altri OB, questi interrompono l'esecuzione dell'OB 1 ed eseguono funzioni specifiche, ad es. per i task di avviamento, la gestione di allarmi ed errori o l'esecuzione di uno specifico codice di programma a particolari intervalli di tempo.
- I blocchi funzionali (FB) sono sottoprogrammi la cui esecuzione viene richiamata da un altro blocco di codice (OB, FB o FC). Il blocco richiamante passa i parametri all'FB e identifica anche un blocco dati (DB) specifico che salva i dati per il richiamo o l'istanza di quell'FB. La possibilità di modificare il DB di istanza consente a un FB generico di comandare il funzionamento di un gruppo di dispositivi. Ad esempio, un unico FB può comandare diverse pompe o valvole utilizzando diversi DB di istanza, ognuno dei quali contiene i parametri di esercizio specifici delle varie pompe o valvole.
- Le funzioni (FC) sono sottoprogrammi la cui esecuzione viene richiamata da un altro blocco di codice (OB, FB o FC). Le FC non sono associate a un DB di istanza e ricevono i parametri dal blocco richiamante. I valori in uscita dalle FC devono essere scritti in un indirizzo di memoria o un DB globale.

Scelta del tipo di struttura del programma utente

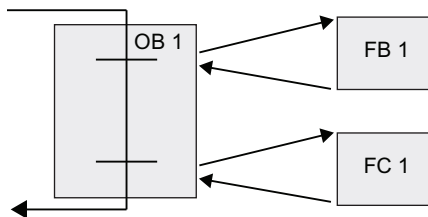
In funzione dei requisiti della propria applicazione si può decidere di creare il programma utente con una struttura lineare o modulare:

- I programmi lineari eseguono tutte le istruzioni per i task di automazione in successione, una dopo l'altra. Generalmente questo tipo di programmi inseriscono tutte le istruzioni nell'OB di esecuzione ciclica (OB 1).
- I programmi modulari richiamano blocchi di codice che eseguono task specifici. Per creare una struttura modulare si deve suddividere il task di automazione in task subordinati, corrispondenti alle funzioni tecnologiche del processo. Ciascun blocco di codice fornisce il segmento di programma per un task subordinato. Per strutturare il programma si richiama uno dei blocchi di codice da un altro blocco.

Struttura lineare:



Struttura modulare:



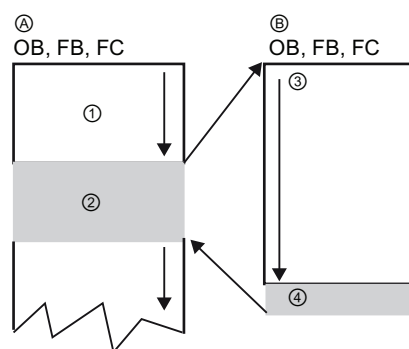
Creando blocchi di codice generici, che possono essere riutilizzati nel programma utente, si semplifica la struttura e l'implementazione del programma utente. L'uso di blocchi di codice generici ha i seguenti vantaggi:

- Si possono creare blocchi di codice riutilizzabili per task standard, ad esempio per comandare una pompa o un motore. Inoltre si possono salvare i blocchi di codice generici in una libreria che può essere utilizzata da applicazioni o soluzioni diverse.
- Scomponendo la struttura del programma utente in componenti modulari collegati a task funzionali il programma risulta più facile da comprendere e gestire. Oltre a consentire di standardizzare la struttura del programma, i componenti modulari permettono di aggiornare e modificare il codice di programma in modo più rapido e semplice.
- I componenti modulari semplificano il test del programma. Strutturando il programma come un insieme di segmenti modulari è possibile testare la funzionalità dei singoli blocchi di codice man mano che li si sviluppa.
- Creando componenti modulari collegati a funzioni tecnologiche specifiche si semplifica e abbrevia la messa in servizio dell'applicazione.

5.3 Utilizzo dei blocchi per la strutturazione del programma

Progettando gli FB e le FC in modo che eseguano task generici si ottengono blocchi di codice modulari. Quindi si struttura il programma facendo in modo che tali blocchi riutilizzabili vengano richiamati da altri blocchi di codice. Il blocco richiamante passa i parametri specifici del dispositivo al blocco richiamato.

- A Blocco richiamante
 B Blocco richiamato (o di interruzione)
 ① Esecuzione del programma
 ② Operazione che richiama un altro blocco
 ③ Esecuzione del programma
 ④ Fine del blocco (per tornare al blocco richiamante)

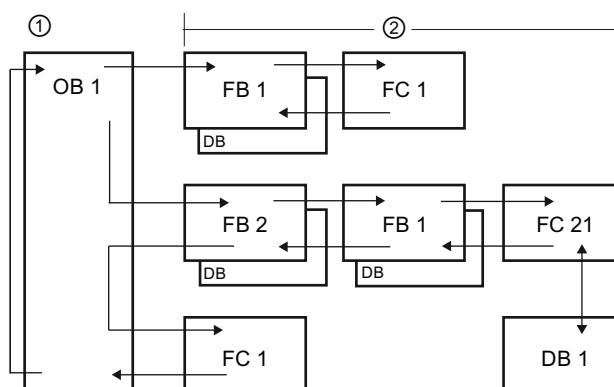


Quando un blocco di codice ne richiama un altro la CPU esegue il codice di programma del blocco richiamato. Terminata l'esecuzione del blocco richiamato la CPU riprende ad eseguire il blocco richiamante.

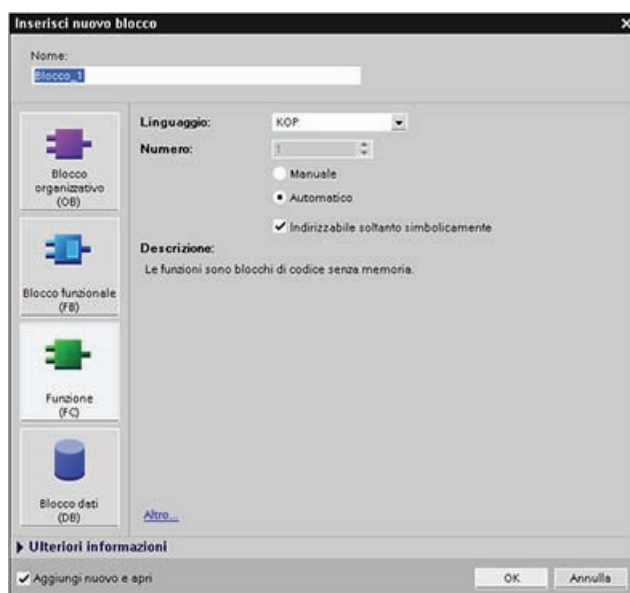
L'elaborazione continua con l'esecuzione dell'istruzione successiva al richiamo del blocco.

Per ottenere una struttura più modulare si possono annidare i richiami.

- ① Avvio del ciclo
- ② Profondità di annidamento



Creazione di blocchi di codice riutilizzabili



Gli OB, gli FB, le FC e i DB globali vengono creati nella finestra di dialogo "Inserisci nuovo blocco" di "Blocchi di programma" nella navigazione di progetto.

Quando si creano i blocchi di codice si deve selezionare il linguaggio di programmazione, mentre per i DB questa operazione non è necessaria perché svolgono solo una funzione di memorizzazione dei dati.

5.3.1 Blocco organizzativo (OB)

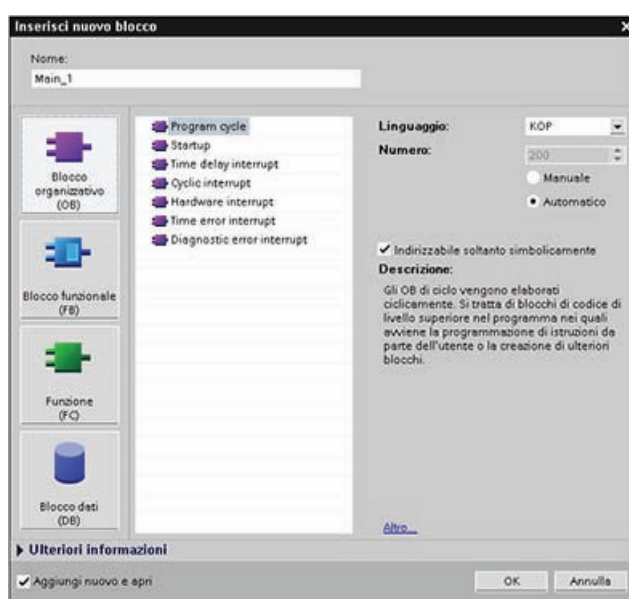
I blocchi organizzativi definiscono la struttura del programma e fungono da interfaccia tra il sistema operativo e il programma utente. Gli OB sono "comandati da eventi", ovvero vengono eseguiti dalla CPU quando si verifica un determinato evento, ad es. un allarme di diagnostica o un intervallo di tempo. Alcuni OB dispongono di eventi di avvio e comportamento predefiniti.

L'OB di ciclo contiene il programma principale. È possibile inserire più di un OB di ciclo nel programma utente. In RUN vengono eseguiti gli OB di ciclo con il livello di priorità inferiore che possono essere interrotti da tutti gli altri tipi di elaborazione del programma. L'OB di avvio non interrompe l'OB di ciclo perché la CPU lo esegue prima di passare in RUN.

Una volta elaborati gli OB di ciclo, la CPU ne riavvia subito l'esecuzione. Questa elaborazione ciclica è quella "normale" dei controllori a logica programmabile. Per molte applicazioni l'intero programma utente è contenuto in un OB di ciclo.

È possibile creare altri OB che eseguono funzioni specifiche, ad es. per i task di avviamento, la gestione di allarmi ed errori o l'esecuzione di uno specifico codice di programma a particolari intervalli di tempo. Questi OB interrompono l'esecuzione degli OB di ciclo del programma.

Per creare nuovi OB per il programma utente si utilizza la finestra di dialogo "Inserisci nuovo blocco".



Un OB può interrompere un altro (sempre che il suo livello di priorità lo consenta). La gestione di queste interruzioni è sempre comandata da evento. Quando si verifica un evento la CPU interrompe l'esecuzione del programma utente e richiama l'OB configurato per elaborare l'evento. Una volta eseguito l'OB la CPU riprende l'esecuzione del programma utente dal punto in cui è stata interrotta.

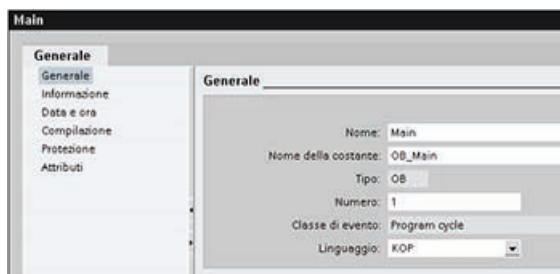
La CPU determina l'ordine di gestione degli eventi di allarme in base alla priorità assegnata a ciascun OB. Ogni evento ha una particolare priorità di elaborazione. Più eventi di allarme possono essere raggruppati in classi di priorità. Per maggiori informazioni consultare il paragrafo relativo all'esecuzione del programma utente nel capitolo Concetti base sui PLC (Pagina 41).

Creazione di un nuovo OB all'interno di una classe di OB

È possibile creare più OB per il programma utente anche all'interno delle classi degli OB di ciclo e di avvio. Creare un OB nella finestra di dialogo "Inserisci nuovo blocco". Specificarne il nome e assegnargli un numero superiore a 200.

Se si creano più OB di ciclo per il programma utente, la CPU li esegue ognuno in base al numero, iniziando da quello principale (che è per default l'OB 1). Ad esempio: al termine del primo OB di ciclo del programma (OB1) la CPU esegue il secondo OB di ciclo (ad es. l'OB 200).

Configurazione dell'esecuzione di un OB



È possibile modificare i parametri di esecuzione di un OB, ad esempio, configurando il tempo di un OB di allarme di ritardo o di un OB di ciclo.

5.3.2 Funzione (FC)

Una funzione (FC) è un blocco di codice che generalmente esegue un'operazione specifica su un gruppo di valori di ingresso. L'FC memorizza i risultati dell'operazione in varie locazioni di memoria.

Le FC consentono di eseguire i seguenti task:

- Operazioni standard riutilizzabili, ad es. per eseguire calcoli matematici.
- Funzioni tecnologiche, ad es. per comandi individuali tramite operazioni di combinazione logica di bit.

Le FC possono essere richiamate anche più volte in punti diversi del programma. La possibilità di riutilizzarle facilita la programmazione dei task che ricorrono frequentemente.

L'FC non è associata a un blocco dati (DB) di istanza, ma scrivono nello stack dei dati locali i dati temporanei per le operazioni di calcolo. I dati temporanei non vengono salvati, per memorizzarli in modo permanente, si deve assegnare il valore di uscita a una locazione di memoria globale, ad es. alla memoria M o a un DB globale.

5.3.3 Blocco funzionale (FB)

Un blocco funzionale (FB) è un blocco di codice che si serve di un blocco dati di istanza per i propri parametri e dati statici. Gli FB dispongono di una memoria per le variabili collocata in un blocco dati (DB) o DB "di istanza". Il DB di istanza mette a disposizione un blocco di memoria che è associato all'istanza (o richiamo) dell'FB e che memorizza i dati al termine dell'esecuzione dell'FB. È possibile associare diversi DB di istanza a diversi richiami dell'FB. Grazie ai DB di istanza è possibile utilizzare un unico FB generico per controllare più dispositivi. Si può realizzare una struttura di programma costituita da un blocco di codice che richiama un FB e un DB di istanza. La CPU esegue il codice di programma nell'FB e memorizza i parametri del blocco e i dati statici locali nel DB di istanza. Quando termina l'esecuzione dell'FB la CPU torna al blocco di codice che ha richiamato l'FB. Il DB di istanza mantiene i valori di quella istanza dell'FB. Questi valori sono disponibili per richiamare successivamente il blocco funzionale nello stesso ciclo di scansione o in altri cicli.

Blocchi di codice riutilizzabili associati a una memoria

Generalmente gli FB vengono utilizzati per controllare l'esecuzione di task o dispositivi che non si esauriscono entro un ciclo di scansione. Per memorizzare i parametri di esercizio in modo che siano rapidamente accessibili da un ciclo di scansione all'altro, ogni FB del programma utente dispone di uno o più DB di istanza. Quando si richiama un FB si specifica anche il DB di istanza che contiene i parametri di blocco e i dati locali statici per quel richiamo o "istanza" dell'FB. Il DB di istanza memorizza questi valori al termine dell'esecuzione dell'FB.

Se un FB viene progettato per task di comando generici è possibile riutilizzarlo per più dispositivi selezionando un diverso DB di istanza per ciascun suo richiamo.

L'FB memorizza i parametri di ingresso (IN), di uscita (OUT) e di ingresso/uscita (IN_OUT) in un DB di istanza.

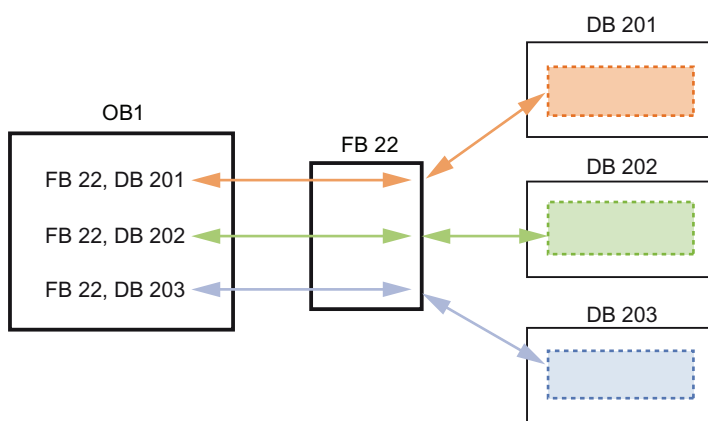
Assegnazione dei valori iniziali

Se non si assegnano valori ai parametri di ingresso, di uscita o di ingresso/uscita di un blocco funzionale (FB), vengono utilizzati i valori memorizzati nel blocco dati di istanza (DB). In alcuni casi l'assegnazione dei valori ai parametri è obbligatoria.

Si possono assegnare ai parametri dell'interfaccia dell'FB dei valori iniziali che verranno trasferiti al DB di istanza associato. Se non si assegna alcun valore, vengono utilizzati quelli memorizzati nel DB di istanza.

Utilizzo di un unico FB con i DB

La seguente figura mostra un OB che richiama per tre volte un FB utilizzando ogni volta un diverso blocco dati. Questa struttura fa sì che un FB generico possa comandare diversi dispositivi simili, ad es. dei motori, assegnando un diverso blocco dati di istanza a ciascun loro richiamo. Ogni DB di istanza memorizza i dati (velocità, tempo della rampa di salita e tempo di funzionamento complessivo) di un particolare dispositivo. Nel presente esempio l'FB 22 controlla tre dispositivi separati e il DB 201 memorizza i dati di esercizio per il primo dispositivo, il DB 202 quelli del secondo dispositivo e il DB 203 quelli del terzo.



5.3.4 Blocco dati (DB)

I blocchi dati (DB) creati per il programma utente consentono di salvare i dati per i blocchi di codice. Tutti i blocchi del programma utente possono accedere ai dati dei DB globali, mentre i DB di istanza memorizzano i dati per blocchi funzionali (FB) specifici. Il DB può essere definito come di sola lettura.

I dati salvati in un DB non vengono cancellati quando termina l'esecuzione del blocco di codice a cui è associato. Si distinguono due tipi di DB:

- I DB globali memorizzano i dati dei blocchi di codice del programma. I dati di un DB globale sono accessibili a qualsiasi OB, FB o FC.
- I DB di istanza memorizzano i dati per un FB specifico. La struttura dei dati di un DB di istanza rispecchia i parametri (Input, Output e InOut) e i dati statici per l'FB (la memoria temporanea per l'FB non viene memorizzata nel DB di istanza).

Nota

Nonostante il DB di istanza rispecchi i dati per un FB specifico, qualsiasi blocco di codice ha la possibilità di accedere ai suoi dati.

5.4 Coerenza dei dati

La CPU mantiene la coerenza tra tutti i dati semplici (ad es. parole o doppie parole) e le strutture definite dal sistema (ad es. IEC_TIMERS o DTL). La lettura o scrittura dei valori non possono essere interrotte (ad es. la CPU protegge l'accesso a un valore di doppia parola finché i suoi quattro byte non sono stati letti o scritti). Per garantire che gli OB di ciclo e di allarme non possano scrivere contemporaneamente nella stessa locazione di memoria, la CPU non esegue l'OB di allarme finché non termina la lettura o la scrittura nell'OB di ciclo.

Se un OB di ciclo e un OB di allarme di un programma utente condividono gli stessi valori di memoria, anche il programma deve garantire che i valori vengano modificati o letti in modo coerente. Per proteggere l'accesso ai valori condivisi si possono inserire nell'OB di ciclo le istruzioni DIS_AIRT e EN_AIRT.

- Inserendo un'istruzione DIS_AIRT nel blocco di codice ci si assicura che l'OB di allarme non possa essere eseguito durante la lettura o la scrittura.
- Inserire le istruzioni che leggono o scrivono i valori che potrebbero essere modificati da un OB di allarme.
- Inserire un'istruzione EN_AIRT alla fine della sequenza per annullare DIS_AIRT e consentire l'esecuzione dell'OB di allarme.

L'esecuzione dell'OB di ciclo può essere interrotta anche dalla richiesta di comunicazione di un dispositivo HMI o di un'altra CPU. Le richieste di comunicazione possono inoltre creare problemi di coerenza dei dati. La CPU garantisce che i tipi di dati semplici vengano sempre letti e scritti in modo coerente dalle istruzioni del programma utente. Poiché il programma utente viene interrotto periodicamente dalla comunicazione, non è possibile garantire che l'HMI aggiorni contemporaneamente valori diversi della CPU. Ad es. i valori visualizzati nel display di un HMI potrebbero appartenere a cicli di scansione diversi della CPU.

Le istruzioni PtP (punto a punto) e PROFINET (ad es. TSEND_C e TRCV_C) trasferiscono buffer di dati che potrebbero essere interrotti. Per garantire la coerenza dei dati dei buffer si deve evitare che vengano eseguite operazioni di lettura e scrittura nei buffer sia nell'OB di ciclo che in quello OB di allarme. Se è necessario modificare i valori del buffer per queste

istruzioni in un OB di allarme, si deve utilizzare un'istruzione DIS_AIRT che posticipi le eventuali interruzioni (un OB di allarme o un allarme di comunicazione da un HMI o un'altra CPU) finché non viene eseguita un'istruzione EN_AIRT.

Nota

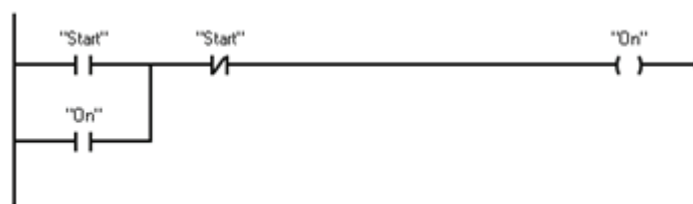
L'istruzione DIS_AIRT ritarda l'elaborazione degli OB di allarme finché non viene eseguita EN_AIRT, che influisce sulla latenza (il tempo che trascorre da un dato evento all'esecuzione dell'OB di allarme) delle interruzioni del programma utente.

5.5 Selezione del linguaggio di programmazione

Come linguaggio di programmazione si può scegliere tra KOP (schema a contatti) o FUP (schema funzionale).

Linguaggio di programmazione KOP

KOP è un linguaggio grafico che consente di rappresentare il programma sotto forma di circuiti elettrici.



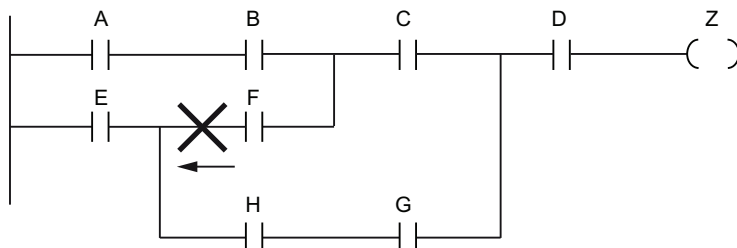
Gli elementi dei circuiti, quali i contatti normalmente chiusi e normalmente aperti e le bobine vengono collegati tra loro per formare dei segmenti (o "network").

Per creare la logica per le operazioni complesse si possono inserire delle diramazioni in modo da realizzare circuiti paralleli. I rami paralleli possono essere aperti verso il basso o collegati direttamente alla barra di alimentazione e si chiudono verso l'alto.

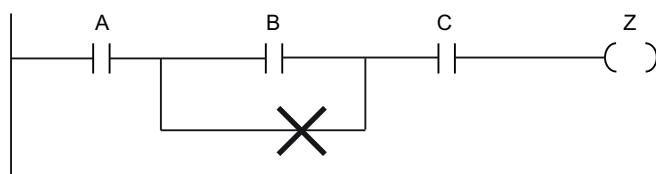
KOP mette a disposizione istruzioni a "box" per svariate funzioni, quali operazioni matematiche, di temporizzazione, di conteggio e di trasferimento.

Quando si crea un segmento KOP è importante tener conto delle seguenti regole:

- Ogni segmento KOP deve terminare con una bobina o un'istruzione a box. Non si possono concludere i segmenti con un'istruzione di confronto o di rilevamento del fronte (positivo o negativo).
- Non è consentito creare rami che possono determinare un'inversione del flusso della corrente.



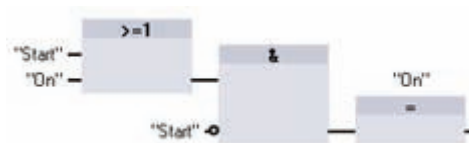
- Non è consentito creare rami che possono provocare un cortocircuito.



Linguaggio di programmazione Schema logico (FUP)

Anche FUP, come KOP, è un linguaggio di programmazione grafico. Per la rappresentazione della logica FUP utilizza i simboli grafici dell'algebra booleana.

Le funzioni matematiche e altre funzioni complesse possono essere rappresentate direttamente tramite i box logici. Per creare la logica per le operazioni complesse si inseriscono rami paralleli tra i box.



EN e ENO per le istruzioni a "box"

Sia KOP che FUP utilizzano il "flusso di corrente" (EN e ENO) per alcune istruzioni a "box". Alcune istruzioni (ad es. quelle matematiche e di trasferimento) visualizzano parametri per EN e ENO. Tali parametri riguardano il flusso della corrente e determinano se l'istruzione verrà eseguita o meno durante un dato ciclo di scansione.

- EN (Enable In) è un ingresso booleano per i box KOP e FUP. La funzione viene eseguita quando l'ingresso EN è attraversato dal flusso di corrente (EN = 1). Se l'ingresso EN di un box KOP è collegato direttamente a sinistra della barra di alimentazione, il box viene sempre eseguito.
- ENO (Enable Out) è un'uscita booleana per i box KOP e FUP. Se l'ingresso EN del box è attraversato dal flusso di corrente e il box esegue la propria funzione senza errori, l'uscita ENO trasmette il flusso di corrente (ENO = 1) all'elemento successivo. Se viene rilevato un errore durante l'esecuzione dell'istruzione a box, il flusso di corrente viene interrotto (ENO = 0) nell'istruzione a box che lo ha generato.

Editor di programma	Ingressi/uscite	Operandi	Tipo di dati
KOP	EN, ENO	Flusso di corrente	BOOL
FUP	EN	I, I:P, Q, M, DB, Temp, flusso di corrente	BOOL
	ENO	Flusso di corrente	BOOL

5.6 Protezione dalla copia

La protezione dalla copia o la protezione del "know-how" consentono di impedire che persone non autorizzate possano accedere a uno o alcuni blocchi di codice (OB, FB o FC) del programma. Creando una password si può limitare l'accesso al blocco di codice.

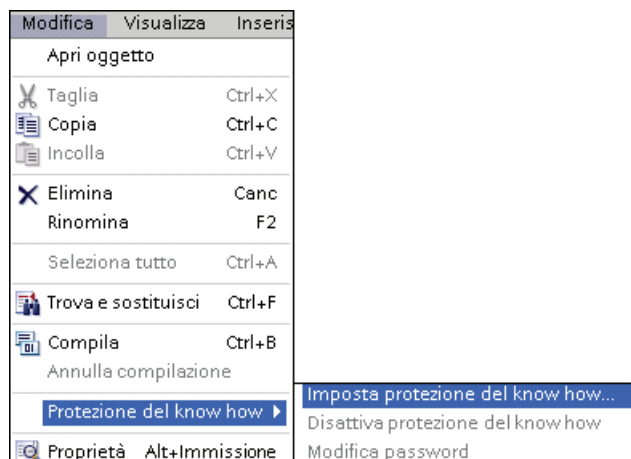
Se si imposta la protezione del "know-how" per un blocco, il codice che vi è contenuto diventa accessibile solo inserendo la password.

Per proteggere il blocco dalla copia selezionare il comando "Protezione KnowHow" nel menu "Modifica".

Specificare quindi la password che consente di accedere al blocco.

La protezione mediante password impedisce alle persone non autorizzate di leggere o modificare il blocco di codice. Se non si dispone della password si possono leggere solo le seguenti informazioni sul blocco di codice:

- Titolo, commento e proprietà del blocco
- Parametri di trasferimento (IN, OUT, IN_OUT, Return)
- Struttura dei richiami del programma
- Variabili globali nei riferimenti incrociati (senza informazioni sul punto di utilizzo, le variabili locali sono nascoste)

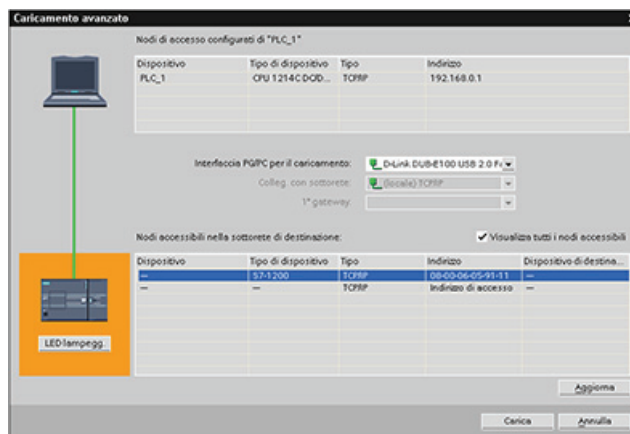


5.7 Caricamento degli elementi del programma nella CPU

Gli elementi del progetto possono essere caricati dal dispositivo di programmazione nella CPU. Quando si carica un progetto la CPU salva il programma utente (OB, FC, FB e DB) nella memoria non volatile.

Il caricamento del progetto dal dispositivo di programmazione nella CPU può essere effettuato da una delle seguenti posizioni:

- "Albero del progetto": fare clic con il tasto destro del mouse sull'elemento del programma quindi selezionare la voce "Carica" nel menu di scelta rapida.
- Menu "Online": fare clic sulla voce "Carica nel dispositivo".
- Barra degli strumenti: fare clic sull'icona "Carica nel dispositivo".



5.8 Caricamento degli elementi del programma dalla CPU

È possibile caricare i blocchi del programma e la tabella delle variabili da una CPU online in un progetto offline, ma non si possono caricare la configurazione dei dispositivi e le tabelle di controllo. Non è possibile effettuare il caricamento in un progetto vuoto, si deve disporre necessariamente di una CPU offline. Non si può caricare un blocco singolo ma solo il programma completo. Quando si effettua un caricamento la CPU offline viene "resettata" (vengono cancellati tutti i blocchi e la tabella delle variabili) dopo una richiesta di conferma. Non è possibile modificare un blocco nell'area online, lo si deve prima caricare nell'area offline, quindi modificarlo e ricaricarlo nel PLC.

Il caricamento può essere effettuato in due modi: mediante drag and drop nell'albero del progetto o sincronizzazione nell'editor di confronto.

Drag and drop nell'albero del progetto

1. Creare un nuovo progetto.
2. Inserirvi una CPU che corrisponda a quella da cui si sta effettuando il caricamento.
3. Espandere una volta il nodo della CPU in modo che compaia la cartella "Blocchi di programma".
4. Nell'albero del progetto espandere il nodo "Accesso online" e il nodo della rete e fare doppio clic su "Aggiorna nodi accessibili".
5. Quando compare l'elenco delle CPU disponibili espandere il nodo di quella rilevante.
6. Fare clic sulla cartella Blocchi di programma dell'area Accesso online e, tenendo premuto il tasto sinistro del mouse, trascinarla nella cartella Blocchi di programma dall'area offline, quindi rilasciare il mouse. Quando ci si trova sull'area esatta il puntatore si trasforma in un '+'.

7. La finestra di dialogo "Anteprima per il caricamento del dispositivo" si apre. Fare clic sulla casella "Continua" e "Carica del dispositivo".
8. Attendere fino al termine del caricamento. Ora l'area offline dovrebbe contenere tutti i blocchi del programma, i blocchi tecnologici e le variabili.
9. Poiché la configurazione del dispositivo non è stata caricata, si devono impostare manualmente le proprietà della CPU, compreso l'indirizzo IP, nella finestra Configurazione dispositivi e inserire gli altri dispositivi nel progetto offline.

Si possono inoltre trascinare gli elementi con drag and drop dall'area online nell'area "Blocchi di programma" di un programma esistente. L'area offline dei blocchi di programma non può cioè essere vuota. In tal caso il programma già presente viene cancellato e sostituito da quello online.

Sincronizzazione nell'editor di confronto

1. Aprire la cartella che contiene il progetto.
2. Nell'albero del progetto selezionare la CPU offline da confrontare.
3. Aprire l'editor "Confronto" facendo clic con il tasto destro del mouse sulla CPU offline o selezionando il comando "Confronta offline/online" nel menu "Strumenti".
4. L'editor elenca le differenze sotto la cartella "Blocchi di programma". Fare clic sul simbolo nella colonna delle azioni. Per caricare il progetto selezionare "Carica dal dispositivo".
5. Fare clic sul pulsante "Sincronizza online e offline" per copiare il progetto dalla CPU online in quella offline.

5.9 Test del programma

Per controllare e modificare i valori del programma utente eseguito da una CPU online si utilizzano le "tabelle di controllo". È possibile creare e salvare nel progetto diverse tabelle di controllo per supportare svariati ambienti di test. In questo modo si possono riprodurre i test durante la messa in servizio o a scopo di assistenza e manutenzione.

Le tabelle di controllo consentono di controllare e interagire con la CPU mentre esegue il programma utente. È possibile visualizzare o modificare i valori non solo per le variabili dei blocchi di codice e dei blocchi dati, ma anche per le aree di memoria della CPU, compresi gli ingressi e le uscite (I e Q), gli ingressi e le uscite della periferia (I:P e Q:P), i merker (M) e i blocchi dati (DB).

La tabella di controllo consente di abilitare le uscite fisiche (Q:P) di una CPU in STOP. La si può usare, ad esempio, per assegnare valori specifici alle uscite mentre si effettua il test del cablaggio per la CPU.

Le tabelle di controllo permettono inoltre di "forzare" una variabile o impostarla su un valore specifico. Per maggiori informazioni sul forzamento vedere il paragrafo relativo al forzamento dei valori nella CPU (Pagina 290) nel capitolo "Online e diagnostica".

Istruzioni di programmazione

6.1 Istruzioni di base

6.1.1 Combinazione logica di bit

Contatti KOP

 Normalm ente aperto	È possibile collegare contatti con altri contatti e realizzare delle combinazioni logiche. Se il bit di ingresso specificato utilizza l'ID di memoria I (ingresso) o Q (uscita), ne viene letto il valore dal registro dell'immagine di processo. I segnali dei contatti fisici del processo di comando sono collegati ai morsetti I del PLC. La CPU effettua la scansione dei segnali degli ingressi collegati e aggiorna ininterrottamente i corrispondenti valori di stato nel registro dell'immagine di processo degli ingressi.
 Normalm ente chiuso	Inserendo ":P" dopo la I è possibile specificare che un dato ingresso fisico deve essere letto direttamente (ad esempio: "%I3.4:P"). In caso di lettura diretta, i valori di dati a bit vengono letti direttamente dall'ingresso fisico invece che dall'immagine di processo. La lettura diretta non implica l'aggiornamento dell'immagine di processo.

Parametro	Tipo di dati	Descrizione
IN	Bool	Bit assegnato

- Il contatto normalmente aperto è chiuso (ON) quando il valore di bit assegnato è uguale a 1.
- Il contatto normalmente chiuso è chiuso (ON) quando il valore di bit assegnato è uguale a 0.
- I contatti collegati in serie creano segmenti logici AND.
- I contatti collegati in parallelo creano segmenti logici OR.

Box FUP, AND, OR e XOR

Nell'programmazione FUP i contatti KOP vengono convertiti in segmenti costituiti da box AND (&), OR (>=1) e OR esclusivo (x), nei quali l'utente può specificare i valori di bit per gli ingressi e le uscite. Collegando i box logici con altri box si possono creare le proprie combinazioni logiche. Una volta inserito un box in un segmento si possono aggiungere altri ingressi con il tool "Inserisci ingresso binario" (selezionarlo nella barra degli strumenti "Preferiti" o nell'albero delle istruzioni e trascinarlo nel lato di ingresso del box). In alternativa si può fare clic con il tasto destro del mouse sul connettore di ingresso del box e selezionare "Inserisci ingresso".

Gli ingressi e le uscite di un box possono essere collegati a un altro box logico oppure, se un ingresso non è collegato, si può specificare un indirizzo o un nome simbolico a bit. Quando l'istruzione a box viene eseguita, gli stati di ingresso attuali vengono applicati alla logica binaria dei box e, se "veri", sarà vera anche l'uscita del box.



Parametro	Tipo di dati	Descrizione
IN1, IN2	Bool	Bit di ingresso

- Perché l'uscita di un box AND sia vera devono essere veri tutti gli ingressi.
- Perché l'uscita di un box OR sia vera deve essere vero un ingresso qualsiasi.
- Perché l'uscita di un box XOR sia vera deve essere vero un numero dispari di ingressi.

Invertitore logico NOT

Nella programmazione in FUP si può selezionare il tool "Nega ingresso binario" nella barra degli strumenti "Preferiti" o nell'albero delle istruzioni e trascinarlo su un ingresso o un'uscita per crearvi un invertitore logico.

—| NOT |—



KOP: Contatto NOT di inversione

FUP: box AND con un ingresso logico invertito

FUP: box AND con ingressi e uscita logici invertiti

Il contatto NOT KOP inverte lo stato logico dell'ingresso del flusso di corrente.

- Se non c'è flusso di corrente in ingresso al contatto NOT, c'è flusso di corrente in uscita.
- Se c'è flusso di corrente in ingresso al contatto NOT, non c'è flusso di corrente in uscita.

Bobina di uscita KOP

"OUT"



Bobina di uscita

"OUT"



Bobina di uscita invertita

L'istruzione bobina di uscita scrive il valore per un bit di uscita. Se il bit di uscita specificato utilizza l'ID di memoria Q, la CPU lo attiva o disattiva nel registro dell'immagine di processo in modo che sia uguale allo stato del flusso di corrente. I segnali di uscita per gli attuatori di comando sono collegati ai morsetti Q dell'S7-1200. In RUN la CPU scansiona ininterrottamente i segnali di ingresso, elabora gli stati degli ingressi in base alla logica del programma e reagisce impostando nuovi valori per gli stati delle uscite nel registro dell'immagine di processo delle uscite. Dopo ciascun ciclo di esecuzione del programma la CPU trasferisce i nuovi stati delle uscite salvati nel registro dell'immagine di processo nei morsetti di uscita cablati.

Inserendo ":P" dopo la Q è possibile specificare che un'uscita fisica venga scritta direttamente (ad esempio: "%Q3.4:P"). In caso di scrittura diretta i valori di dati di bit vengono scritti nell'immagine di processo delle uscite e direttamente nell'uscita fisica.

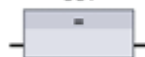
Parametro	Tipo di dati	Descrizione
OUT	Bool	Bit assegnato

- Se la bobina di uscita è attraversata dal flusso di corrente il bit di uscita viene impostato a 1.
- Se la bobina di uscita non è attraversata dal flusso di corrente il bit di uscita viene impostato a 0.
- Se una bobina di uscita invertita è attraversata dal flusso di corrente il bit di uscita viene impostato a 0.
- Se una bobina di uscita invertita non è attraversata dal flusso di corrente il bit di uscita viene impostato a 1.

Box di assegnazione dell'uscita FUP

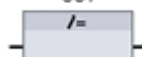
Nella programmazione FUP, le bobine KOP vengono trasformate in box di assegnazione (= e /=) nei quali si specifica un indirizzo di bit per l'uscita. È possibile collegare gli ingressi e le uscite dei box alla logica degli altri box o specificare un indirizzo di bit.

"OUT"



Assegnazione dell'uscita

"OUT"



Assegnazione dell'uscita invertita

"OUT"



Assegnazione dell'uscita con uscita invertita

Parametro	Tipo di dati	Descrizione
OUT	Bool	Bit assegnato

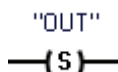
- Se l'ingresso del box di uscita è 1 il bit OUT viene impostato a 1.
- Se l'ingresso del box di uscita è 0 il bit OUT viene impostato a 0.
- Se l'ingresso del box di uscita invertita è 1 il bit OUT viene impostato a 0.
- Se l'ingresso del box di uscita invertita è 0 il bit OUT viene impostato a 1.

6.1.1.1 Istruzioni di impostazione e reset

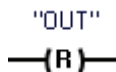
S e R: Imposta e Resetta 1 bit

- Quando S (Imposta) è attiva, il valore di dati nell'indirizzo OUT viene impostato a 1. Quando S è disattivata OUT resta invariato.
- Quando R (Resetta) è attiva, il valore di dati nell'indirizzo OUT viene impostato a 0. Quando R è disattivata OUT resta invariato.
- Queste istruzioni possono essere inserite in qualsiasi punto del segmento.

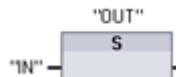
KOP: Imposta



KOP: Resetta



FUP: Imposta



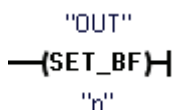
FUP: Resetta



Parametro	Tipo di dati	Descrizione
IN (o collegamento ai contatti/porte logiche)	Bool	Indirizzo di bit da controllare
OUT	Bool	Indirizzo di bit da impostare o resettare

SET_BF e RESET_BF: Imposta e Resetta campo di bit

KOP: SET_BF



KOP: RESET_BF



FUP: SET_BF



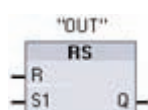
FUP: RESET_BF



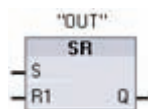
Parametro	Tipo di dati	Descrizione
n	Costante	Numero di bit da scrivere
OUT	Elemento di un array booleano	Elemento iniziale di un campo di bit da impostare o resettare Esempio: #MyArray[3]

- Quando SET_BF è attiva, viene assegnato il valore di dati 1 a "n" bit a partire dall'indirizzo OUT. Quando SET_BF è disattivata OUT resta invariato.
- RESET_BF scrive il valore di dati 0 in "n" bit a partire dall'indirizzo OUT. Quando RESET_BF è disattivata OUT resta invariato.
- Queste istruzioni devono essere inserite nell'ultima posizione a destra del ramo.

RS e SR: Latch di bit con set e reset dominante



RS è un latch in cui è dominante set. Se i segnali di impostazione (S1) e di reset (R) sono entrambi veri, l'indirizzo di uscita OUT sarà 1.



SR è un latch in cui è dominante reset. Se i segnali di impostazione (S) e di reset (R1) sono entrambi veri, l'indirizzo di uscita OUT sarà 0.

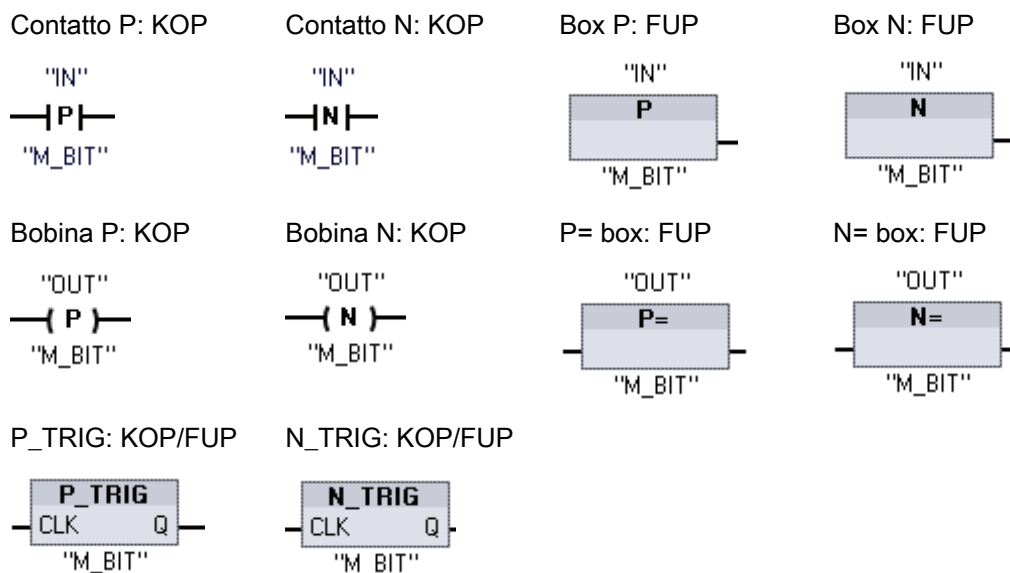
Il parametro OUT specifica l'indirizzo di bit che viene impostato o resettato. L'uscita OUT opzionale Q rispecchia lo stato del segnale dell'indirizzo "OUT".

Parametro	Tipo di dati	Descrizione
S, S1	BOOL	Ingresso di impostazione (set); 1 indica che è dominante
R, R1	BOOL	Ingresso di reset; 1 indica che è dominante
OUT	BOOL	Uscita del bit assegnato "OUT"
Q	BOOL	Segue lo stato del bit "OUT"

Istruzione	S1	R	Bit "OUT"
RS	0	0	Stato precedente
	0	1	0
	1	0	1
	1	1	1
SR	S	R1	
	0	0	Stato precedente
	0	1	0
	1	0	1
	1	1	0

6.1.1.2 Istruzioni di fronte di salita e di discesa

Istruzioni di rilevamento di transizione positiva e negativa



Parametro	Tipo di dati	Descrizione
M_BIT	Bool	Merker in cui è stato salvato lo stato precedente dell'ingresso
IN	Bool	Bit di ingresso di cui deve essere rilevato il fronte di transizione
OUT	Bool	Bit di uscita che indica che è stato rilevato un fronte di transizione
CLK	Bool	Bit di flusso di corrente o di ingresso di cui deve essere rilevato il fronte di transizione
Q	Bool	Uscita che indica che è stato rilevato un fronte

- Contatto P: KOP** Lo stato di questo contatto è vero quando viene rilevata una transizione positiva (da OFF a ON) nel bit assegnato "IN". Lo stato logico del contatto viene quindi combinato con lo stato del flusso di corrente in ingresso per impostare lo stato del flusso di corrente in uscita. Il contatto P può essere inserito in qualsiasi punto del segmento tranne che alla fine del ramo.
- Contatto N: KOP** Lo stato di questo contatto è vero quando viene rilevata una transizione negativa (da ON a OFF) nel bit di ingresso assegnato. Lo stato logico del contatto viene quindi combinato con lo stato del flusso di corrente in ingresso per impostare lo stato del flusso di corrente in uscita. Il contatto N può essere inserito in qualsiasi punto del segmento tranne che alla fine del ramo.
- Box P: FUP** Lo stato logico dell'uscita è vero quando viene rilevata una transizione positiva (da OFF a ON) nel bit di ingresso assegnato. Il box P può essere inserito solo all'inizio di un ramo.

Box N: FUP	Lo stato logico dell'uscita è vero quando viene rilevata una transizione negativa (da ON a OFF) nel bit di ingresso assegnato. Il box N può essere inserito solo all'inizio di un ramo.
Bobina P: KOP	Il bit assegnato "OUT" è vero quando viene rilevata una transizione positiva (da OFF a ON) nel flusso di corrente in ingresso alla bobina. Lo stato del flusso di corrente in ingresso passa sempre attraverso la bobina come il flusso di corrente in uscita. La bobina P può essere inserita in qualsiasi punto del segmento.
Bobina N: KOP	Il bit assegnato "OUT" è vero quando viene rilevata una transizione negativa (da ON a OFF) nel flusso di corrente in ingresso alla bobina. Lo stato del flusso di corrente in ingresso passa sempre attraverso la bobina come il flusso di corrente in uscita. La bobina N può essere inserita in qualsiasi punto del segmento.
P= box: FUP	Il bit assegnato "OUT" è vero quando viene rilevata una transizione positiva (da OFF a ON) nello stato logico della connessione di ingresso del box oppure, se il box si trova all'inizio del ramo, nell'assegnazione del bit di ingresso. Lo stato logico dell'ingresso passa sempre attraverso il box come lo stato logico dell'uscita. Il box P= può essere inserito in qualsiasi punto del ramo.
N= box: FUP	Il bit assegnato "OUT" è vero quando viene rilevata una transizione negativa (da ON a OFF) nello stato logico della connessione di ingresso del box oppure, se il box si trova all'inizio del ramo, nell'assegnazione del bit di ingresso. Lo stato logico dell'ingresso passa sempre attraverso il box come lo stato logico dell'uscita. Il box N= può essere inserito in qualsiasi punto del ramo.
P_TRIG: KOP/FUP	Il flusso di corrente o lo stato logico dell'uscita Q sono veri quando viene rilevata una transizione positiva (da OFF a ON) nello stato dell'ingresso CLK (FUP) o nel flusso di corrente in ingresso CLK (KOP). In KOP l'istruzione P_TRIG non può essere inserita all'inizio o alla fine di un segmento. In FUP l'istruzione P_TRIG può essere inserita in qualsiasi punto tranne che alla fine di un ramo.
N_TRIG (KOP/FUP)	Il flusso di corrente o lo stato logico dell'uscita Q sono veri quando viene rilevata una transizione negativa (da ON a OFF) nello stato dell'ingresso CLK (FUP) o nel flusso di corrente in ingresso CLK (KOP). In KOP l'istruzione N_TRIG non può essere inserita all'inizio o alla fine di un segmento. In FUP l'istruzione P_TRIG può essere inserita in qualsiasi punto tranne che alla fine di un ramo.

Tutte le istruzioni di fronte utilizzano un merker (M_BIT) per memorizzare lo stato precedente del segnale di ingresso controllato. Il fronte viene rilevato confrontando lo stato dell'ingresso con quello del merker. Se gli stati indicano che l'ingresso è cambiato nella direzione rilevante, viene rilevato un fronte e l'uscita diventa vera. In caso contrario l'uscita diventa falsa.

Nota

Le istruzioni con i fronti valutano i valori dell'ingresso e del merker ad ogni esecuzione, compresa la prima. Quando si progetta il programma è necessario tener conto degli stati dell'ingresso e del merker per consentire o meno il rilevamento dei fronti nel primo ciclo di scansione.

Poiché il merker deve essere mantenuto in memoria da un'esecuzione all'altra, si deve utilizzare un univoco bit per ciascuna istruzione di fronte e non utilizzare lo stesso bit in altri punti del programma. Si deve inoltre evitare di usare la memoria temporanea e le aree di memoria che possono essere influenzate da altre funzioni di sistema, ad es. da un aggiornamento degli I/O. Per l'assegnazione degli M_BIT (in un DB di istanza) utilizzare solo la memoria M, i DB globali o la memoria statica.

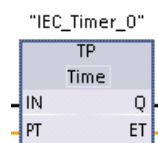
6.1.2 Temporizzatori

Le istruzioni di temporizzazione vengono utilizzate per creare ritardi programmati:

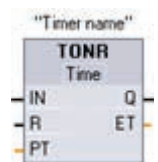
- TP: il Temporizzatore come impulso genera un impulso con una durata preimpostata.
- TON: l'uscita Q Temporizzatore come ritardo all'inserzione viene impostata su ON al termine di un tempo di ritardo preimpostato.
- TOF: l'uscita Q Temporizzatore come ritardo alla disinserzione viene impostata su OFF al termine di un tempo di ritardo preimpostato.
- TONR: l'uscita Temporizzatore come ritardo all'inserzione con memoria viene impostata su ON al termine di un tempo di ritardo preimpostato. Il tempo trascorso viene accumulato per più periodi di temporizzazione finché non viene resettato dall'ingresso R.
- RT: resetta un temporizzatore cancellando i dati di temporizzazione memorizzati nel DB di istanza del temporizzatore specificato.

Per salvare i propri dati ciascun temporizzatore utilizza una struttura memorizzata in un blocco dati, che viene assegnato quando si inserisce l'istruzione nell'editor.

Quando si inseriscono le istruzioni di temporizzazione in un blocco funzionale si può scegliere l'opzione "blocco dati di multiistanza"; in questo modo i nomi delle strutture dei temporizzatori possono essere diversi e contrassegnare strutture di dati diverse, ma i dati di temporizzazione sono contenuti in un unico blocco dati, per cui non è necessario creare un DB separato per ogni temporizzatore. Si riduce così il tempo di elaborazione e la memoria di dati necessaria per la gestione dei temporizzatori. Le strutture dei dati di temporizzazione contenute nel blocco dati di multiistanza condiviso non interagiscono tra loro.



I temporizzatori TP, TON e TOF hanno gli stessi parametri di ingresso e di uscita.



Il temporizzatore TONR ha anche il parametro di ingresso di reset R.

Definire un "Nome del temporizzatore" che definisca il blocco dati e descriva la funzione del temporizzatore nel processo.

"Nome del
temporizzatore"

L'istruzione RT resetta i dati di temporizzazione per il temporizzatore in oggetto.

----[RT]----

Parametro	Tipo di dati	Descrizione
IN	Bool	Attiva l'ingresso del temporizzatore
R	Bool	Resetta a zero il tempo trascorso nel TONR
PT	Bool	Ingresso del valore di tempo preimpostato
Q	Bool	Uscita del temporizzatore
ET	Time	Uscita del valore del tempo trascorso
Blocco dati del temporizzatore	DB	Specificare il temporizzatore che deve essere resettato dall'istruzione RT

Il parametro IN avvia e arresta i temporizzatori:

- La commutazione da 0 a 1 del parametro IN avvia i temporizzatori TP, TON e TONR.
- La commutazione da 1 a 0 del parametro IN avvia il temporizzatore TOF.

La seguente tabella illustra le conseguenze delle variazioni del valore dei parametri PT e IN.

Timer	Variazioni nei parametri PT e IN
TP	• La variazione di PT non ha alcuna conseguenza durante l'esecuzione del temporizzatore. • La variazione di IN non ha alcuna conseguenza durante l'esecuzione del temporizzatore.
TON	• La variazione di PT non ha alcuna conseguenza durante l'esecuzione del temporizzatore. • Se IN diventa falso durante l'esecuzione del temporizzatore il temporizzatore viene resettato e arrestato.

Timer	Variazioni nei parametri PT e IN
TOF	- La variazione di PT non ha alcuna conseguenza durante l'esecuzione del temporizzatore. - Se IN diventa vero durante l'esecuzione del temporizzatore il temporizzatore viene resettato e arrestato.
TONR	- La variazione di PT non ha alcuna conseguenza durante l'esecuzione del temporizzatore, ma ne ha quando l'esecuzione riprende. - Se IN diventa falso durante l'esecuzione del temporizzatore il temporizzatore viene arrestato ma non resettato. Se IN diventa di nuovo vero il temporizzatore avvia la temporizzazione a partire dal valore di tempo accumulato.

Valori di TIME

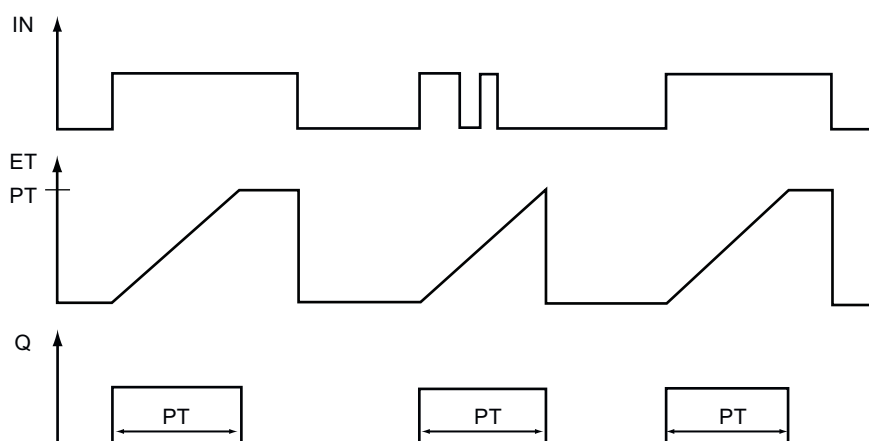
I valori di PT (tempo preimpostato) ed ET (tempo trascorso) vengono salvati nella memoria come numeri interi a 32 bit con segno che rappresentano i millisecondi. I dati di TIME utilizzano l'ID T# e possono essere specificati come unità di tempo semplice "T#200ms" o composta "T#2s_200ms".

Tipo di dati	Dimensione	Campi numerici validi
TIME	32 bit	Da T#-24d_20h_31m_23s_648ms a T#24d_20h_31m_23s_647ms
	Salvati come	Da -2.147.483.648 ms a +2.147.483.647 ms

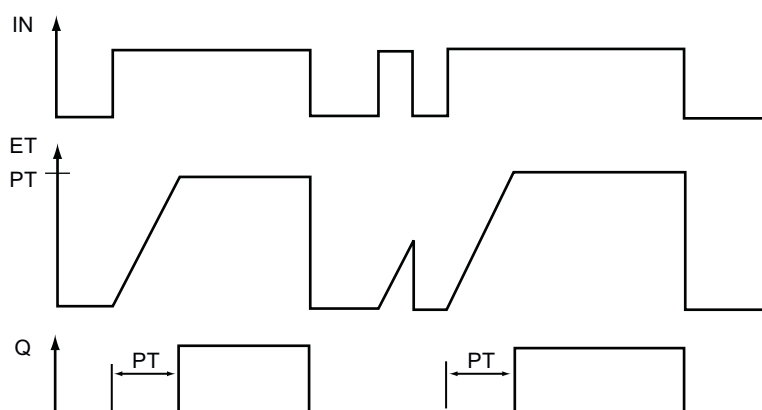
Nota

Il campo negativo del tipo di dati TIME sopra indicato non è utilizzabile con le istruzioni di temporizzazione. I valori PT (tempo preimpostato) negativi vengono impostati a zero quando viene eseguita l'istruzione di temporizzazione. ET (tempo preimpostato) è sempre un valore positivo.

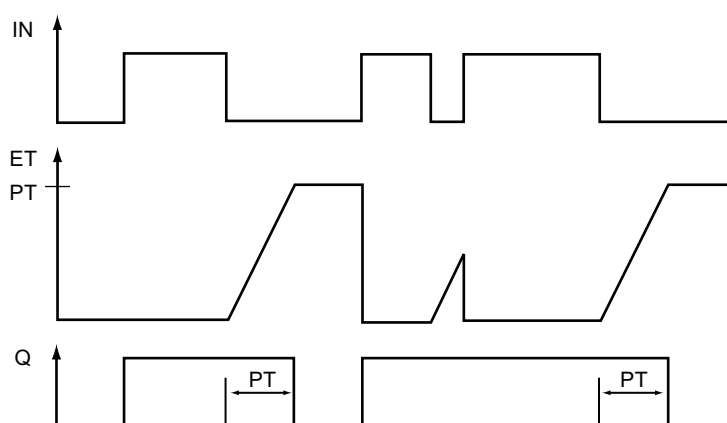
TP:
Diagramma di
un
temporizzatore
e come
impulso



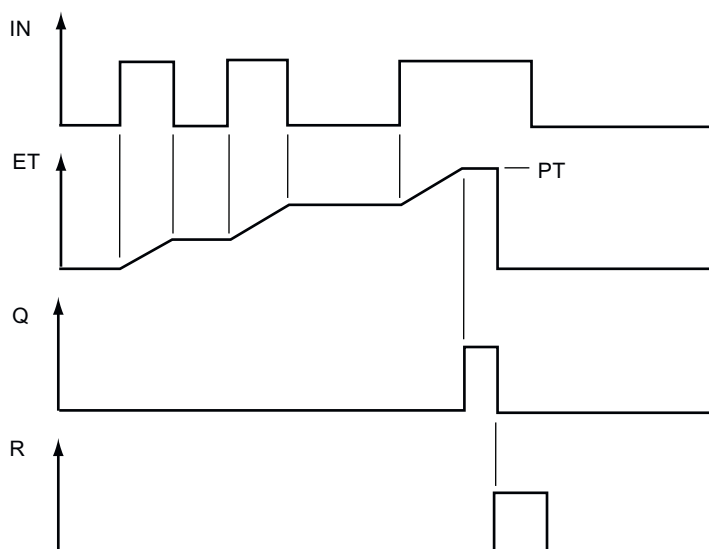
TON:
Diagramma di
un
temporizzatore
e come ritardo
all'inserzione



TOF:
Diagramma di
un
temporizzatore
e come ritardo
alla
disinserzione



TONR:
Diagramma di
un
temporizzatore
e come ritardo
all'inserzione
con memoria



6.1.3 Contatori

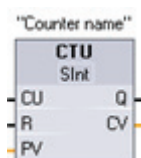
6.1.3.1 Contatori

Le istruzioni di conteggio consentono di contare gli eventi interni del programma e quelli esterni del processo:

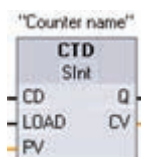
- CTU è un contatore di conteggio in avanti.
- CTD è un contatore di conteggio all'indietro.
- CTUD è un contatore di conteggio in avanti e all'indietro.

Per salvare i propri dati ciascun contatore utilizza una struttura memorizzata in un blocco dati che viene assegnato quando si inserisce l'istruzione nell'editor. Queste istruzioni utilizzano contatori software la cui velocità massima di conteggio è limitata dalla velocità di esecuzione dell'OB in cui sono stati inseriti. L'OB in cui si trovano le istruzioni deve essere eseguito abbastanza spesso da rilevare tutte le transizioni degli ingressi CU o CD. Per informazioni sulle operazioni di conteggio veloce vedere l'istruzione CTRL_HSC.

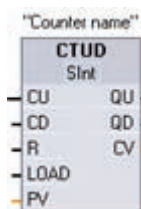
Quando si inseriscono le istruzioni di conteggio in un blocco funzionale si può scegliere l'opzione "blocco dati di multiistanza"; in questo modo i nomi delle strutture dei contatori possono essere diversi e contrassegnare strutture di dati diverse, ma i dati di conteggio sono contenuti in un unico blocco dati, per cui non è necessario creare un DB separato per ogni contatore. Si riduce così il tempo di elaborazione e la memoria dei dati necessaria per i contatori. Le strutture dei dati di conteggio contenute nel blocco dati di multiistanza condiviso non interagiscono tra loro.



Selezionare il tipo di dati del valore di conteggio nell'elenco a discesa sotto il nome del box.



Creare un "Nome del contatore" che definisca il blocco dati e descriva la funzione del contatore nel processo.

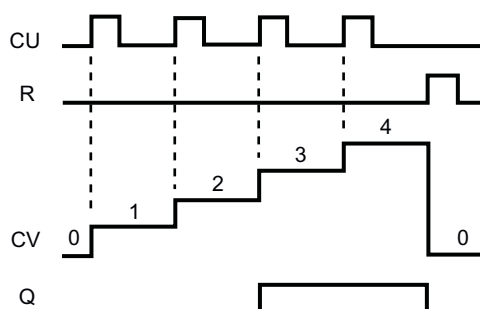


Parametro	Tipo di dati	Descrizione
CU, CD	Bool	Conta in avanti o indietro di uno
R (CTU, CTUD)	Bool	Resetta a zero il valore di conteggio
LOAD (CTD, CTUD)	Bool	Carica il controllo per il valore preimpostato

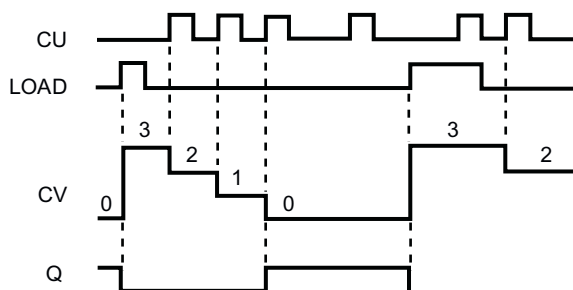
Parametro	Tipo di dati	Descrizione
PV	SInt, Int, DInt, USInt, UInt, UDIInt	Valore di conteggio preimpostato
Q, QU	Bool	Vero se $CV \geq PV$
QD	Bool	Vero se $CV \leq 0$
CV	SInt, Int, DInt, USInt, UInt, UDIInt	Valore di conteggio attuale

Il campo numerico dei valori di conteggio varia in funzione del tipo di dati selezionato. Se il valore è un numero intero senza segno è possibile contare all'indietro fino a zero o in avanti fino al limite del campo. Se il valore è un numero intero con segno è possibile contare all'indietro fino al limite negativo del campo e in avanti fino al limite positivo.

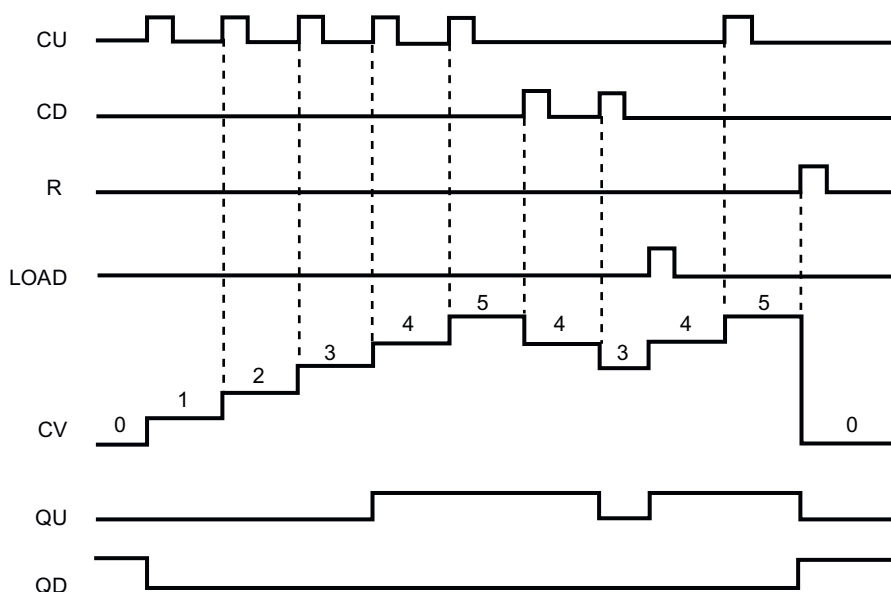
CTU: CTU conta in avanti di 1 quando il valore del parametro CU cambia da 0 a 1. Se il valore del parametro CV (valore di conteggio attuale) è maggiore o uguale al valore del parametro PV (valore di conteggio preimpostato), il parametro di uscita del contatore Q = 1. Se il valore del parametro di reset R cambia da 0 a 1 il valore di conteggio attuale viene resettato a 0. La seguente figura mostra un diagramma di temporizzazione CTU con un valore di conteggio costituito da un numero intero senza segno (dove $PV = 3$).



CTD: CTD conta indietro di 1 quando il valore del parametro CD cambia da 0 a 1. Se il valore del parametro CV (valore di conteggio attuale) è uguale o inferiore a 0, il parametro di uscita del contatore Q = 1. Se il valore del parametro LOAD cambia da 0 a 1, il valore del parametro PV (valore preimpostato) viene caricato nel contatore come nuovo CV (valore di conteggio attuale). La seguente figura mostra un diagramma di temporizzazione CTD con un valore di conteggio costituito da un numero intero senza segno (dove $PV = 3$).



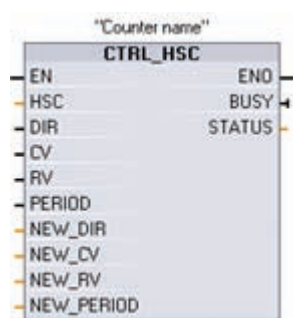
CTUD: CTUD conta in avanti o indietro di 1 quando gli ingressi di conteggio in avanti (CU) o all'indietro (CD) passano da 0 a 1. Se il valore del parametro CV (valore di conteggio attuale) è uguale o maggiore del valore del parametro PV (valore preimpostato) il parametro di uscita del contatore QU = 1. Se il valore del parametro CV è inferiore o uguale a zero il parametro di uscita del contatore QD = 1. Se il valore del parametro LOAD cambia da 0 a 1, il valore del parametro PV (valore preimpostato) viene caricato nel contatore come nuovo CV (valore di conteggio attuale). Se il valore del parametro di reset R cambia da 0 a 1, il valore di conteggio attuale viene resettato a 0. La seguente figura mostra un diagramma di temporizzazione CTUD con un valore di conteggio costituito da un numero intero senza segno (dove PV = 4).



6.1.3.2 Istruzione CTRL_HSC

L'istruzione CTRL_HSC comanda i contatori veloci utilizzati per contare gli eventi che si verificano più rapidamente della velocità di esecuzione dell'OB. La velocità di conteggio delle istruzioni CTU, CTD e CTUD è limitata dalla velocità di esecuzione dell'OB in cui sono state inserite. Per informazioni sulle velocità di conteggio massime degli HSC consultare gli schemi elettrici della CPU riportati nei dati tecnici (Pagina 300).

Un uso tipico dei contatori veloci è il conteggio degli impulsi generati da un encoder per il comando della rotazione di un albero.



Per salvare i propri dati ciascuna istruzione CTRL_HSC utilizza una struttura memorizzata in un blocco dati che viene assegnato quando si inserisce l'istruzione nell'editor.

Creare un "Nome del contatore" che definisca il blocco dati e descriva la funzione del contatore nel processo.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
HSC	IN	HW_HSC	Identificatore dell'HSC
DIR	IN	Bool	1 = richiesta della nuova direzione
CV	IN	Bool	1 = richiesta di impostare il nuovo valore di conteggio
RV	IN	Bool	1 = richiesta di impostare il nuovo valore di riferimento
PERIOD	IN	Bool	1 = richiesta di impostare il nuovo valore del periodo (solo per la modalità di misura della frequenza)
NEW_DIR	IN	Int	Nuova direzione: 1= in avanti -1= indietro
NEW_CV	IN	DInt	Nuovo valore di conteggio
NEW_RV	IN	DInt	Nuovo valore di riferimento
NEW_PERIOD	IN	Int	Nuovo valore del periodo in secondi: ,01; ,1 o 1 (solo per la modalità di misura della frequenza)
BUSY	OUT	Bool	Funzione occupata
STATUS	OUT	WORD	Codice della condizione di esecuzione

Per poter utilizzare i contatori veloci nel programma è innanzitutto necessario configurarli in Configurazione dispositivi nelle impostazioni del progetto. Le impostazioni per la configurazione degli HSC consentono di selezionare i modi di conteggio, i collegamenti di I/O, l'assegnazione degli allarmi e il funzionamento come contatore veloce o come dispositivo di misura della frequenza degli impulsi. L'esecuzione dei contatori veloci può essere comandata o meno dal programma.

La maggior parte dei parametri di configurazione dei contatori veloci possono essere impostati solamente nella configurazione dispositivi del progetto. Alcuni vengono tuttavia inizializzati nella configurazione ma possono essere modificati in seguito mediante comandi del programma.

I parametri dell'istruzione CTRL_HSC consentono di comandare il processo di conteggio tramite il programma:

- Impostare la direzione di conteggio sul valore NEW_DIR
- Impostare il valore di conteggio attuale sul valore NEW_CV

- Impostare il valore di riferimento sul valore NEW_RV
- Impostare il valore di Period (per la modalità di misura della frequenza) sul valore NEW_PERIOD

Se i seguenti valori di merker booleani vengono impostati a 1 durante l'esecuzione di CTRL_HSC, il corrispondente valore NEW_xxx viene caricato nel contatore. Se sono presenti più richieste (vengono impostati più merker contemporaneamente), vengono elaborate in una singola esecuzione dell'istruzione CTRL_HSC.

- DIR = 1 è la richiesta di caricare il valore NEW_DIR, 0 = nessun cambiamento
- CV = 1 è la richiesta di caricare il valore NEW_CV, 0 = nessun cambiamento
- RV = 1 è la richiesta di caricare il valore NEW_RV, 0 = nessun cambiamento
- PERIOD = 1 è la richiesta di caricare il valore NEW_PERIOD, 0 = nessun cambiamento

L'istruzione CTRL_HSC viene generalmente inserita in un OB di interrupt di processo che viene eseguito in seguito all'attivazione dell'evento di interrupt di processo del contatore. Se, ad esempio, un evento CV=RV attiva l'interrupt del contatore, un OB di interrupt di processo esegue l'istruzione CTRL_HSC e può modificare il valore di riferimento caricando un valore NEW_RV.

Il valore di conteggio attuale non è disponibile nei parametri CTRL_HSC. L'indirizzo dell'immagine di processo per la memorizzazione del valore di conteggio attuale viene assegnato durante la configurazione hardware del contatore veloce. È possibile utilizzare la logica del programma per leggere direttamente il valore di conteggio; il valore restituito al programma sarà il conteggio corretto per l'istante in cui è stato letto il contatore, il quale continuerà a contare gli eventi veloci. Il valore di conteggio attuale potrebbe quindi cambiare prima che il programma termini un processo utilizzando un precedente valore di conteggio.

Particolarità dei parametri di CTRL_HSC:

- Se non viene richiesto l'aggiornamento del valore di un parametro i corrispondenti valori di ingresso vengono ignorati.
- Il parametro DIR è valido solo se la direzione di conteggio configurata è impostata su "Programma utente (comando direzione interno)". Il modo in cui si utilizza questo parametro può essere definito nella configurazione dei dispositivi degli HSC.
- Per gli HSC S7-1200 nella CPU o nella signal board il parametro BUSY ha sempre il valore 0.

Codici delle condizioni di errore: Se si verifica un errore ENO viene impostato a 0 e l'uscita STATUS contiene il codice della condizione di errore.

Valore di STATUS (W#16#...)	Descrizione
0	Nessun errore
80A1	L'identificatore dell'HSC non indirizza un HSC
80B1	Valore non ammesso in NEW_DIR
80B2	Valore non ammesso in NEW_CV
80B3	Valore non ammesso in NEW_RV
80B4	Valore non ammesso in NEW_PERIOD

6.1.3.3 Funzionamento del contatore veloce

Un contatore veloce (HSC) può essere utilizzato come ingresso per un encoder incrementale. L'encoder incrementale fornisce un numero specifico di impulsi di rotazione, oltre a un impulso di reset che interviene una volta per giro. I clock e l'impulso di reset dell'encoder incrementale forniscono gli ingressi per l'HSC.

Il primo di diversi valori di preimpostazione viene caricato nell'HSC. Le uscite vengono attivate per il periodo di tempo in cui il valore attuale è minore di quello di preimpostazione. L'HSC fornisce un interrupt quando il valore attuale è uguale a quello di preimpostazione, quando si verifica un reset e anche nel caso di un cambio di direzione.

Ogni volta che si verifica l'evento di interrupt "valore di conteggio attuale = valore di preimpostazione", viene caricato un nuovo valore di preimpostazione e viene impostato lo stato successivo delle uscite. Se si verifica un evento di interrupt perché viene resettato il contatore, vengono impostati il primo valore di preimpostazione e i primi stati di segnale delle uscite e viene ripetuto il ciclo.

Poiché gli interrupt si verificano ad una velocità molto più bassa di quella di conteggio degli HSC, è possibile controllare con precisione le operazioni ad alta velocità con un impatto relativamente basso sul ciclo del controllore programmabile. Grazie alla possibilità di assegnare degli interrupt, è possibile caricare ciascun nuovo valore di preimpostazione in una routine di interrupt separata semplificando il controllo dello stato (in alternativa è possibile elaborare tutti gli eventi di interrupt in un'unica routine di interrupt).

Selezione delle funzioni per l'HSC

In un dato modo di conteggio tutti gli HSC operano nella stessa maniera. I modi operativi di base sono 4:

- Contatore a una fase con controllo interno della direzione
- Contatore a una fase con controllo esterno della direzione
- Contatore a due fasi con 2 ingressi di clock
- Contatore con fasi A/B in quadratura

Ogni tipo di HSC può essere utilizzato con o senza ingresso di reset. Se si attiva l'ingresso di reset (con le limitazioni indicate nella seguente tabella) il valore attuale viene resettato finché non si disattiva l'ingresso.

- Funzione di frequenza: Alcuni modi degli HSC consentono di configurare gli HSC (tipo di conteggio) in modo che rilevino la frequenza anziché il valore attuale di conteggio degli impulsi. Esistono tre diversi periodi di misura della frequenza: 0,01, 0,1, o 1,0 secondi.

Il periodo di misura della frequenza determina la frequenza con cui l'HSC calcola e rileva un nuovo valore. La frequenza rilevata è un valore medio determinato dal numero totale di impulsi nell'ultimo periodo di misura. Se la frequenza cambia rapidamente, il valore rilevato sarà un valore intermedio tra la frequenza maggiore e quella minore che si è verificata durante il periodo di misura. La frequenza viene sempre indicata in Hertz (impulsi al secondo) a prescindere dall'impostazione del periodo di misura della stessa.

- Modi e ingressi dei contatori: La tabella seguente riepiloga gli ingressi utilizzati per le funzioni di clock, di controllo della direzione e di reset associate agli HSC.

Nonostante non sia possibile utilizzare un ingresso per due diverse funzioni, gli ingressi non utilizzati dall'attuale modo di un HSC possono essere destinati ad un utilizzo diverso. Se ad es. un HSC1 è in un modo che utilizza ingressi integrati ma non il reset esterno (I0.3), I0.3 può essere impiegato per allarmi di fronte o per HSC2.

Descrizione			Assegnazione di default degli ingressi			Funzione
HSC	HSC1	Integrato o signal board o monitor PTO 0 ¹	I0.0 I4.0 PTO 0 Impulso	I0.1 I4.1 PTO 0 Direzione	I0.3 I4.3 -	
	HSC:	Built in o signal board o PTO di controllo 1 ¹	I0.2 I4.2 PTO 1 Impulsi	I0.3 I4.3 PTO 1 Direzione	I0.1 I4.1 -	
	HSC3 ²	Integrato	I0.4	I0.5	I0.7	
	HSC4 ³	Integrato	I0.6	I0.7	I0.5	
	HSC5 ⁴	Integrato o signal board	I1.0 I4.0	I1.1 I4.1	I1.2 I4.3	
	HSC6 ⁴	Built in o signal board	I1.3 I4.2	I1.4 I4.3	I1.5 I4.1	
Modo	Contatore a una fase con controllo interno della direzione		Clock	-	-	Impulso o frequenza
					Resetta	Impulso
	Contatore a una fase con controllo esterno della direzione		Clock	Direzione	-	Impulso o frequenza
					Resetta	Impulso
	Contatore a due fasi con 2 ingressi di clock		Clock in avanti	Clock indietro	-	Impulso o frequenza
					Resetta	Impulso
	Contatore con fasi A/B in quadratura		Fase A	Fase B	-	Impulso o frequenza
					Fase Z	Impulso
	Monitor delle uscite di treni di impulsi (PTO) ¹		Clock	Direzione	-	Impulso

¹ Il controllo delle uscite di treni di impulsi utilizza sempre clock e direzione. Se la relativa uscita PTO è configurata solo per impulso, l'uscita di direzione dovrebbe normalmente essere impostata per il conteggio positivo.

² HSC3 con un ingresso di rest non è ammesso per la CPU 1211C che supporta solo 6 ingressi integrati.

³ HSC4 non è ammesso per la CPU 1211C che supporta solo 6 ingressi integrati.

⁴ HSC5 e HSC6 sono supportati soltanto dalla CPU 1211C e dalla CPU 1212C se è installata una signal board.

Accesso al valore attuale dell'HSC

La CPU memorizza il valore attuale di ogni HSC in un indirizzo di ingresso (I). La seguente tabella mostra gli indirizzi di default assegnati al valore attuale di ogni HSC. Modificando le proprietà della CPU nella Configurazione dispositivi è possibile cambiare l'indirizzo I del valore attuale.

Contatore veloci	Tipo di dati	Indirizzo di default
HSC1	DInt	ID1000
HSC2	DInt	ID1004
HSC3	DInt	ID1008
HSC4	DInt	ID1012
HSC5	DInt	ID1016
HSC6	DInt	ID1020

Impossibile forzare gli I/O digitali assegnati ai dispositivi HSC

Gli I/O digitali utilizzati dai dispositivi HSC vengono assegnati durante la configurazione dei dispositivi. Quando sono assegnati a questi dispositivi indirizzi di I/O digitali, i valori di tali indirizzi non possono essere modificati dalla funzione di forzamento nella tabella di controllo.

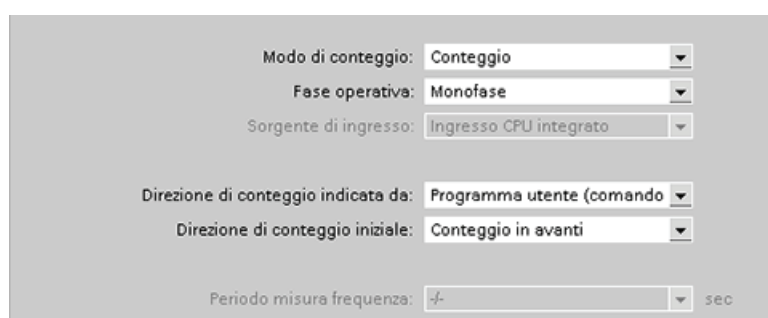
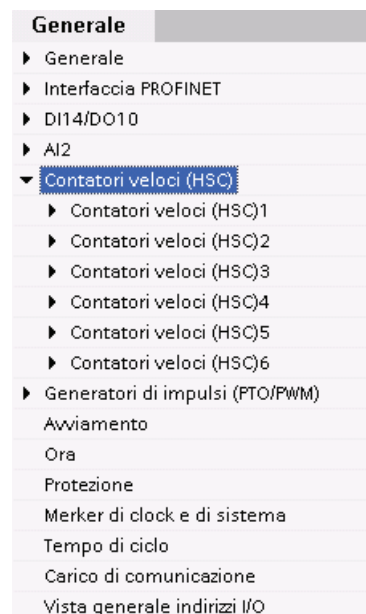
6.1.3.4 Configurazione dell'HSC

La CPU permette all'utente di configurare fino a 6 contatori veloci. I parametri di ogni singolo HSC possono essere configurati nelle "Proprietà" della CPU.

Configurare i parametri dei contatori veloci modificando le "Proprietà" della CPU.

Dopo aver abilitato l'HSC, configurare gli altri parametri, ad es. la funzione di conteggio, i valori iniziali, le opzioni di reset e gli eventi di interrupt.

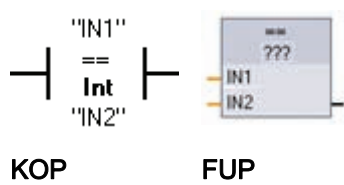
Dopo aver configurato l'HSC, utilizzare l'istruzione CTRL_HSC nel programma per comandare il funzionamento dell'HSC.



Resetta valori
Valore di conteggio iniziale: 0
Valore di riferimento iniziale: 0

Opzioni di resettaggio
☒ Questo HSC deve impiegare un ingresso di resettaggio esterno. Il resettaggio comporta la cancellazione del valore attuale.
Resetta livello seg
☒ Genera allarme per evento con valore di conteggio uguale al valore di riferimento.:
Nome evento: Valore di conteggio uguale
Interrupt di processo: --
☐ Genera allarme per evento di resettaggio esterno.:
Nome evento:
Interrupt di processo: --
☐ Attiva allarme per evento di cambio direzione.:
Nome evento:
Interrupt di processo: --

6.1.4 Confronto



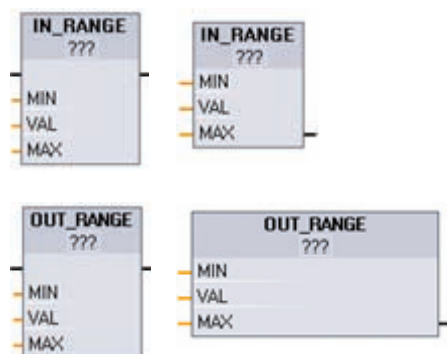
Le istruzioni di confronto consentono di confrontare due valori con lo stesso tipo di dati. Se il confronto del contatto KOP è vero il contatto viene attivato. Se il confronto del box FUP è vero l'uscita del box è vera.

Fare clic sull'istruzione nell'editor di programma e selezionare il tipo di confronto e il tipo di dati nei menu a discesa.

Tipo di relazione	Il confronto è vero se:
==	IN1 è uguale a IN2
<>	IN1 è diverso da IN2
>=	IN1 è maggiore di o uguale a IN2
<=	IN1 è inferiore o uguale a IN2
>	IN1 è maggiore di IN2
<	IN1 è inferiore a IN2

Parametro	Tipo di dati	Descrizione
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, String, Char, Time, DTL, costante	Valori da confrontare

Istruzioni Valore compreso nel campo e Valore fuori campo



KOP

FUP

Le istruzioni IN_RANGE e OUT_RANGE consentono di testare se un valore di ingresso si trova entro o al di fuori di un campo di valori specificato. Se il confronto è vero l'uscita del box è vera.

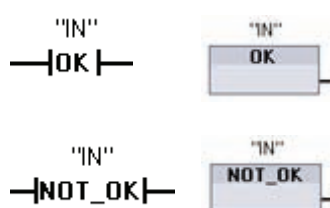
I parametri di ingresso MIN, VAL e MAX devono avere lo stesso tipo di dati.

Fare clic sull'istruzione nell'editor di programma e selezionare il tipo di dati nei menu a discesa.

Tipo di relazione	Il confronto è vero se:
IN_RANGE	$\text{MIN} \leq \text{VAL} \leq \text{MAX}$
OUT_RANGE	$\text{VAL} < \text{MIN}$ o $\text{VAL} > \text{MAX}$

Parametro	Tipo di dati	Descrizione
MIN, VAL, MAX	SInt, Int, DInt, USInt, UInt, UDIInt, Real, costante	Ingressi del comparatore

Istruzioni Verifica validità e Verifica nullità



KOP

FUP

Le istruzioni OK e NOT_OK consentono di verificare se un riferimento ai dati di ingresso è o meno un numero reale valido secondo IEEE specifica 754. Se il confronto del contatto KOP è vero il contatto viene attivato e fa passare il flusso di corrente. Se il box FUP è vero l'uscita del box è vera.

Un valore Real o LReal non è valido se è +/- INF (infinito), NaN (non è un numero) o se è un valore denormalizzato. Un valore denormalizzato è un numero molto vicino allo zero. Nei calcoli la CPU lo sostituisce con uno zero.

Istruzione	Il test del numero Real è vero se:
OK	Il valore di ingresso è un numero Real
NOT_OK	Il valore di ingresso non è un numero Real

Parametro	Tipo di dati	Descrizione
IN	Real, LReal	Dati di ingresso

6.1.5 Funzioni matematiche

Istruzioni Somma, Sottrai, Moltiplica e Dividi



Le funzioni matematiche a box consentono di programmare le principali operazioni matematiche:

- ADD: somma ($IN1 + IN2 = OUT$)
- SUB: sottrazione ($IN1 - IN2 = OUT$)
- MUL: moltiplicazione ($IN1 * IN2 = OUT$)
- DIV: divisione ($IN1 / IN2 = OUT$)

Le divisioni di numeri interi troncano la parte frazionaria del quoziente per fornire un numero intero in uscita.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Nota

I parametri principali della funzione matematica IN1, IN2 e OUT devono avere lo stesso tipo di dati.

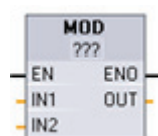
Parametro	Tipo di dati	Descrizione
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, costante	Ingressi dell'operazione matematica
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Uscita dell'operazione matematica

Se attiva ($EN = 1$) la funzione matematica esegue l'operazione specificata sui valori di ingresso (IN1 e IN2) e salva il risultato nell'indirizzo di memoria specificato dal parametro di uscita (OUT). Una volta eseguita correttamente l'operazione, l'istruzione imposta $ENO = 1$.

Stato di ENO	Descrizione
1	Nessun errore
0	Il risultato dell'operazione matematica non è compreso entro il campo numerico valido del tipo di dati selezionato. Viene restituita la parte meno significativa del risultato corrispondente alla dimensione di destinazione.

Stato di ENO	Descrizione
0	Divisione per 0 (IN2 = 0): Il risultato non è definito e viene restituito zero.
0	Real/LReal: se uno dei valori di ingresso è NAN (non è un numero) viene restituito NAN.
0	ADD Real/LReal: se entrambi i valori IN sono INF con segni diversi l'operazione non è valida e viene restituito NaN.
0	SUB Real/LReal: se entrambi i valori IN sono INF con lo stesso segno l'operazione non è valida e viene restituito NaN.
0	MUL Real/LReal: se un valore IN è zero e l'altro è INF l'operazione non è valida e viene restituito NaN.
0	DIV Real/LReal: se entrambi i valori IN sono zero o INF l'operazione non è valida e viene restituito NaN.

6.1.5.1 Istruzione MOD



L'istruzione MOD (modulo) viene utilizzata per l'operazione matematica $IN1 \text{ modulo } IN2$. L'operazione $IN1 \text{ MOD } IN2 = IN1 - (IN1 / IN2) =$ parametro OUT.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

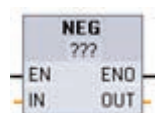
Nota

I parametri IN1, IN2 e OUT devono avere lo stesso tipo di dati.

Parametro	Tipo di dati	Descrizione
IN1 e IN2	Int, DInt, USInt, UInt, UDIInt, costante	Ingressi del modulo
OUT	Int, DInt, USInt, UInt, UDIInt	Uscita del modulo

Stato di ENO	Descrizione
1	Nessun errore
0	Il valore $IN2 = 0$, ad OUT viene assegnato il valore zero

Istruzione NEG



L'istruzione NEG (negazione) consente di invertire il segno aritmetico del valore nel parametro IN e di salvare il risultato nel parametro OUT.

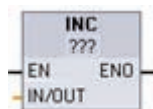
Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Nota

I parametri IN e OUT devono avere lo stesso tipo di dati.

Parametro	Tipo di dati	Descrizione
IN	SInt, Int, DInt, Real, LReal, costante	Ingresso dell'operazione matematica
OUT	SInt, Int, DInt, Real, LReal	Uscita dell'operazione matematica

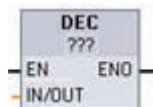
Stato di ENO	Descrizione
1	Nessun errore
0	Il valore risultante non è compreso entro il campo numerico valido del tipo di dati selezionato. Esempio per SInt: NEG (-128) dà come risultato +128 che è maggiore del valore massimo consentito per il tipo di dati.

Istruzioni Incrementa e Decrementa

Le istruzioni INC e DEC consentono di:

- Incrementa un valore di numero intero con o senza segno

INC (incrementa): Parametro IN/OUT valore +1 = parametro IN/OUT valore



- Decrementa un valore di numero intero con o senza segno

DEC (decrementa): Parametro IN/OUT valore - 1 = parametro IN/OUT valore

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Parametro	Tipo di dati	Descrizione
IN/OUT	SInt, Int, DInt, USInt, UInt, UDInt	Ingresso e uscita dell'operazione matematica

Stato di ENO	Descrizione
1	Nessun errore
0	Il valore risultante non è compreso entro il campo numerico valido del tipo di dati selezionato. Esempio per SInt: INC (127) dà come risultato -128 che è maggiore del valore massimo consentito per il tipo di dati.

Istruzione Genera valore assoluto



L'istruzione ABS consente di leggere il valore assoluto di un numero intero o reale con segno nel parametro IN e di salvare il risultato nel parametro OUT.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

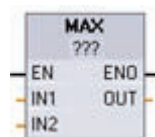
Nota

I parametri IN e OUT devono avere lo stesso tipo di dati.

Parametro	Tipo di dati	Descrizione
IN	SInt, Int, DInt, Real, LReal	Ingresso dell'operazione matematica
OUT	SInt, Int, DInt, Real, LReal	Uscita dell'operazione matematica

Stato di ENO	Descrizione
1	Nessun errore
0	Il risultato dell'operazione matematica non è compreso entro il campo numerico valido del tipo di dati selezionato. Esempio per SInt: ABS (-128) dà come risultato +128 che è maggiore del valore massimo consentito per il tipo di dati.

Istruzioni MIN e MAX



Le istruzioni MIN (minimo) e MAX (massimo) consentono di effettuare le operazioni descritte di seguito.

- MIN confronta il valore di due parametri IN1 e IN2 e assegna il valore minimo (il più basso) al parametro OUT.
- MAX confronta il valore di due parametri IN1 e IN2 e assegna il valore massimo (il più alto) al parametro OUT.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Nota

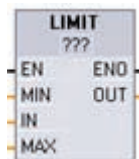
I parametri IN1, IN2 e OUT devono avere lo stesso tipo di dati.

Parametro	Tipo di dati	Descrizione
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, Costante	Ingressi dell'operazione matematica
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real	Uscita dell'operazione matematica

Stato di ENO	Descrizione
1	Nessun errore
0	Solo per il tipo di dati Real: • Uno o entrambi gli ingressi non è un numero Real (NaN). • L'OUT risultante è +/- INF (infinito).

Istruzione Imposta valore limite

L'istruzione Imposta valore limite consente di verificare se il valore del parametro IN è compreso entro il campo di valori specificato dai parametri MIN e MAX. Se il valore di IN non è compreso entro il campo, il valore di OUT viene bloccato sul valore MIN o MAX.



- Se il valore del parametro IN è compreso entro il campo specificato il valore di IN viene salvato nel parametro OUT.
- Se il valore del parametro IN non è compreso entro il campo specificato, il valore OUT corrisponde al valore del parametro MIN (se il valore di IN è inferiore a quello di MIN) oppure al valore del parametro MAX (se il valore di IN è superiore a quello di MAX).

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Nota

I parametri MIN, IN, MAX e OUT devono avere lo stesso tipo di dati.

Parametro	Tipo di dati	Descrizione
MIN, IN e MAX	SInt, Int, DInt, USInt, UInt, UDIInt, Real, costante	Ingressi dell'operazione matematica
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real	Uscita dell'operazione matematica

Stato di ENO	Descrizione
1	Nessun errore
0	Real: se uno o alcuni valori per MIN, IN e MAX non è un numero (NaN) viene restituito NaN.
0	Se MIN è maggiore di MAX il valore di IN viene assegnato a OUT.

Funzioni matematiche in virgola mobile

Le funzioni matematiche in virgola mobile consentono di programmare le operazioni matematiche utilizzando il tipo di dati Real o LReal:

- SQR: Genera quadrato ($IN^2 = OUT$)
- SQRT: Genera radice quadrata ($\sqrt{IN} = OUT$)
- LN: Genera logaritmo naturale ($LN(IN) = OUT$)
- EXP: Genera valore esponenziale ($e^{IN} = OUT$), dove la base $e = 2,71828182845904523536$
- SIN: Genera valore del seno ($\sin(\text{radianti } IN) = OUT$)
- COS: Genera valore del coseno ($\cos(\text{radianti } IN) = OUT$)
- TAN: Genera valore della tangente ($\tan(\text{radianti } IN) = OUT$)
- ASIN: Genera valore dell'arcoseno ($\arcsin(IN) = \text{radianti } OUT$), dove $\sin(\text{radianti } OUT) = IN$
- ACOS: Genera valore dell'arcocoseno ($\arccos(IN) = \text{radianti } OUT$), dove $\cos(\text{radianti } OUT) = IN$
- ATAN: Genera valore dell'arcotangente ($\arctan(IN) = \text{radianti } OUT$), dove $\tan(\text{radianti } OUT) = IN$
- FRAC: Rileva i decimali (parte frazionale di un numero in virgola mobile $IN = OUT$)
- EXPT: Calcola la potenza ($IN1^{IN2} = OUT$)



Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa. I parametri di EXPT IN1 e OUT sono sempre Real. Il tipo di dati per il parametro dell'esponente IN2 può essere selezionato.



Parametro	Tipo di dati	Descrizione
IN, IN1	Real, LReal, costante	Ingressi
IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, costante	Ingresso dell'esponente EXPT
OUT	Real, LReal	Uscite

Stato di ENO	Istruzione	Condizione	Risultato (OUT)
1	Tutti	Nessun errore	Risultato valido
0	SQR	Il risultato è maggiore del campo Real/LReal valido	+INF
		IN è +/- NaN (non è un numero)	+NaN
	SQRT	IN è negativo	-NaN
		IN è +/- INF (infinito) o +/- NaN	+/- INF o +/- NaN
	LN	IN è 0,0, negativo, -INF o -NaN	-NaN
		IN è +INF o +NaN	+INF o +NaN
	EXP	Il risultato è maggiore del campo Real/LReal valido	+INF
		IN è +/- NaN	+/- NaN
	SIN, COS, TAN	IN è +/- INF o +/- NaN	+/- INF o +/- NaN
	ASIN, ACOS	IN non è compreso nel campo valido da -1,0 a +1,0	+NaN
		IN è +/- NaN	+/- NaN
	ATAN	IN è +/- NaN	+/- NaN
	FRAC	IN è +/- INF o +/- NaN	+NaN
	EXPT	IN1 è +INF e IN2 non è -INF	+INF
		IN1 è negativo o -INF	+NaN se IN2 è Real/LReal, -INF negli altri casi
		IN1 o IN2 è +/- NaN	+NaN
		IN1 è 0,0 e IN2 è Real/LReal (solamente)	+NaN

6.1.6 Operazioni di trasferimento

Istruzioni Copia valore e Copia area



Le istruzioni di trasferimento consentono di copiare degli elementi di dati in un nuovo indirizzo di memoria e di convertirli da un tipo di dati in un altro. Il trasferimento non determina la modifica dei dati di origine.

- MOVE: copia in un nuovo indirizzo un elemento di dati memorizzato in un indirizzo specificato
- MOVE_BLK: trasferimento con interruzione che copia un blocco di elementi di dati in un nuovo indirizzo
- UMOVE_BLK: trasferimento senza interruzione che copia un blocco di elementi di dati in un nuovo indirizzo

MOVE		
Parametro	Tipo di dati	Descrizione
IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Char, Array, Struct, DTL, Time	Indirizzo di origine
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord, Char, Array, Struct, DTL, Time	Indirizzo di destinazione

MOVE_BLK, UMOVE_BLK		
Parametro	Tipo di dati	Descrizione
IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, Byte, Word, DWord	Indirizzo iniziale di origine
COUNT	UInt	Numero di elementi di dati da copiare
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, Byte, Word, DWord	Indirizzo iniziale di destinazione

Nota

Regole per le operazioni di copia dei dati

- Per copiare il tipo di dati Bool utilizzare SET_BF, RESET_BF, R, S o la bobina di uscita (KOP)
- Per copiare un unico tipo di dati semplice utilizzare MOVE
- Per copiare un array di tipi di dati semplici utilizzare MOVE_BLK o UMOVE_BLK
- Per copiare una struttura utilizzare MOVE
- Per copiare una stringa utilizzare S_CONV
- Per copiare il carattere di una stringa utilizzare MOVE
- Le istruzioni MOVE_BLK e UMOVE_BLK non consentono di copiare array o strutture nelle aree di memoria I, Q o M.

L'istruzione MOVE copia un unico elemento di dati dall'indirizzo di origine specificato dal parametro IN nell'indirizzo di destinazione specificato dal parametro OUT.

Le istruzioni MOVE_BLK e UMOVE_BLK hanno anche un parametro COUNT che specifica quanti elementi di dati vengono copiati. Il numero di byte copiati per gli elementi dipende dal tipo di dati assegnati ai nomi delle variabili dei parametri IN e OUT nella tabella delle variabili PLC.

Le istruzioni MOVE_BLK e UMOVE_BLK si differenziano per la modalità di gestione degli allarmi:

- Durante l'esecuzione di MOVE_BLK gli eventi di allarme vengono **messi in coda ed elaborati**. Utilizzare l'istruzione MOVE_BLK nei casi in cui i dati contenuti nell'indirizzo di destinazione non vengono usati in un sottoprogramma di OB di allarme oppure, in casi di utilizzo, se non è necessario che siano coerenti. Se l'operazione MOVE_BLK viene interrotta l'ultimo elemento di dati trasferito nell'indirizzo di destinazione è completo e coerente. L'operazione MOVE_BLK viene ripresa al termine dell'esecuzione dell'OB di allarme.
- Gli eventi di allarme vengono **messi in coda ma non elaborati** finché non termina l'esecuzione di UMOVE_BLK. Utilizzare l'istruzione UMOVE_BLK nei casi in cui, per poter eseguire un sottoprogramma di OB di allarme, è necessario che l'operazione di

trasferimento sia terminata e che i dati di destinazione siano coerenti. Per maggiori informazioni consultare il capitolo sulla coerenza dei dati (Pagina 94).

Dopo l'esecuzione dell'istruzione MOVE ENO è sempre vera.

Stato di ENO	Condizione	Risultato
1	Nessun errore	Tutti gli elementi COUNT sono stati copiati correttamente
0	Il campo di origine (IN) o di destinazione (OUT) sono maggiori dell'area di memoria disponibile	Gli elementi con le dimensioni adatte vengono copiati. Gli elementi parziali non vengono copiati.

Istruzioni di predefinitone della memoria



Le istruzioni FILL_BLK e UFILL_BLK consentono di:

- FILL_BLK: l'istruzione Inserire i dati nel campo predefinisce un campo di indirizzo con copie di un elemento di dati specificato.
- UFILL_BLK: l'istruzione Compila campo senza interruzione predefinisce un campo di indirizzo con copie di un elemento di dati specificato.

Parametro	Tipo di dati	Descrizione
IN	SInt, Int, DIntT, USInt, UInt, UDIInt, Real, BYTE, Word, DWord	Indirizzo di origine dei dati
COUNT	USInt, UInt	Numero di elementi di dati da copiare
OUT	SInt, Int, DIntT, USInt, UInt, UDIInt, Real, BYTE, Word, DWord	Indirizzo di destinazione dei dati

Nota

Regole per le operazioni di predefinitone della memoria

- Per effettuare la predefinitone con il tipo di dati BOOL utilizzare SET_BF, RESET_BF, R, S o la bobina di uscita (KOP)
- Per effettuare la predefinitone con un unico tipo di dati semplice utilizzare MOVE
- Per predefinire un array con un tipo di dati semplici utilizzare FILL_BLK o UFILL_BLK
- Per predefinire il carattere di una stringa utilizzare MOVE
- Le istruzioni FILL_BLK e UFILL_BLK non consentono di predefinire array nelle aree di memoria I, Q o M.

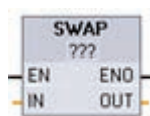
Le istruzioni FILL_BLK e UFILL_BLK copiano l'elemento dati di origine IN nella destinazione in cui il parametro OUT specifica l'indirizzo iniziale. L'operazione di copia viene ripetuta e un blocco ininterrotto di indirizzi viene predefinito finché il numero di copie non diventa uguale al parametro COUNT.

Le istruzioni FILL_BLK e UFILL_BLK si differenziano per la modalità di gestione degli allarmi:

- Durante l'esecuzione di FILL_BLK gli eventi di allarme vengono **messi in coda ed elaborati**. Utilizzare l'istruzione FILL_BLK nei casi in cui i dati contenuti nell'indirizzo di destinazione non vengono usati in un sottoprogramma di OB di allarme oppure, in casi di utilizzo, se non è necessario che siano coerenti.
- Gli eventi di allarme vengono **messi in coda ma non elaborati** finché non termina l'esecuzione di UFILL_BLK. Utilizzare l'istruzione UFILL_BLK nei casi in cui, per poter eseguire un sottoprogramma di OB di allarme, è necessario che l'operazione di trasferimento sia terminata e che i dati di destinazione siano coerenti.

Stato di ENO	Condizione	Risultato
1	Nessun errore	L'elemento IN è stato copiato correttamente in tutte le destinazioni COUNT
0	Il campo di destinazione (OUT) è maggiore dell'area di memoria disponibile	Gli elementi con le dimensioni adatte vengono copiati. Gli elementi parziali non vengono copiati.

6.1.6.1 Istruzione Modifica disposizione



L'istruzione SWAP consente di invertire l'ordine dei byte all'interno di elementi di dati costituiti da due o quattro byte, lasciando tuttavia invariato l'ordine dei bit nei byte. Dopo l'esecuzione dell'istruzione SWAP ENO è sempre vero.

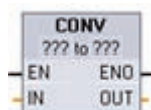
Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Parametro	Tipo di dati	Descrizione
IN	Word, DWord	Byte di dati IN disposti in ordine
OUT	Word, DWord	Byte di dati OUT disposti in ordine inverso

	Esempio: parametro IN = MB0, prima dell'esecuzione di SWAP				Esempio: parametro OUT = MB4, dopo l'esecuzione di SWAP			
Indirizzo	MB0	MB1			MB4	MB5		
W#16#1234	12	34			34	12		
WORD	MSB	LSB			MSB	LSB		
Indirizzo	MB0	MB1	MB2	MB3	MB4	MB5	MB6	MB7
DW#16#12345678	12	34	56	78	78	56	34	12
DWORD	MSB			LSB	MSB			LSB

6.1.7 Conversione

Istruzione Converti valore



L'istruzione CONVERT consente di convertire un elemento di dati da un tipo di dati in un altro. Fare clic sotto il nome del box e selezionare i tipi di dati IN e OUT nel menu a discesa.

Dopo che è stato selezionato il tipo di dati (da convertire) l'elenco a discesa (dei tipi in cui convertire) visualizza una lista di possibili conversioni. Le conversioni da e verso BCD16 sono limitate al tipo di dati Int. Le conversioni da e verso BCD32 sono limitate al tipo di dati DInt.

Fare clic sotto il nome del box e selezionare i tipi di dati nei menu a discesa.

Parametro	Tipo di dati	Descrizione
IN	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord, Real, LReal, Bcd16, Bcd32	Valore IN
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord, Real, LReal, Bcd16, Bcd32	Valore IN convertito in un nuovo tipo di dati

Stato di ENO	Descrizione	Risultato (OUT)
1	Nessun errore	Risultato valido
0	IN è +/- INF o +/- NaN	+/- INF o +/- NaN
0	Il risultato è maggiore del campo valido per il tipo di dati OUT	OUT è impostato sui byte meno significativi di IN

Istruzioni Arrotonda numero e Genera numero intero



ROUND converte un numero reale in un numero intero. I decimali del numero reale vengono arrotondati al numero intero successivo (IEEE - round to nearest). Se la cifra decimale del numero reale è esattamente la metà della differenza tra due numeri interi (cioè 10,5), il numero reale viene arrotondato all'intero pari. Ad esempio ROUND (10,5) = 10 or ROUND (11,5) = 12.



TRUNC converte un numero reale in numero intero. La parte frazionaria del numero reale viene troncata a zero (IEEE - round to zero).

Parametro	Tipo di dati	Descrizione
IN	Real, LReal	Ingresso in virgola mobile
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Uscita arrotondata o troncata

Stato di ENO	Descrizione	Risultato (OUT)
1	Nessun errore	Risultato valido
0	IN è +/- INF o +/- NaN	+/- INF o +/- NaN

Istruzioni Genera da un numero in virgola mobile il numero intero superiore e Genera da un numero in virgola mobile il numero intero inferiore



CEIL converte un numero reale nel più piccolo numero intero maggiore o uguale al numero reale (IEEE - round to +infinity).



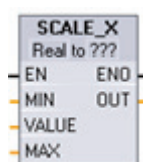
FLOOR converte un numero reale nel più grande numero intero inferiore o uguale al numero reale (IEEE - round to -infinity).

Parametro	Tipo di dati	Descrizione
IN	Real, LReal	Ingresso in virgola mobile
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Uscita convertita

Stato di ENO	Descrizione	Risultato (OUT)
1	Nessun errore	Risultato valido
0	IN è +/- INF o +/- NaN	+/- INF o +/- NaN

6.1.7.1 Istruzioni Riporta in scala e Normazione

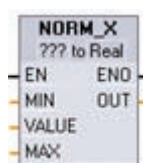
Istruzioni Riporta in scala e Normazione



SCALE_X riporta in scala il parametro VALUE, costituito da un numero reale normalizzato, dove ($0,0 \leq \text{VALUE} \leq 1,0$) nel tipo di dati e nel campo di valori specificati dai parametri MIN e MAX:

$$\text{OUT} = \text{VALUE} (\text{MAX} - \text{MIN}) + \text{MIN}$$

Nell'istruzione SCALE_X, i parametri MIN, MAX e OUT devono avere lo stesso tipo di dati.



NORM_X normalizza il parametro VALUE entro il campo di valori specificato dai parametri MIN e MAX:

$$\text{OUT} = (\text{VALUE} - \text{MIN}) / (\text{MAX} - \text{MIN}), \text{ dove } (0,0 \leq \text{OUT} \leq 1,0)$$

Nell'istruzione NORM_X, i parametri MIN, VALUE e MAX devono avere lo stesso tipo di dati.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Parametro	Tipo di dati	Descrizione
MIN	SInt, Int, DInt, USInt, UInt, UDIInt, Real	Valore di ingresso minimo per il campo
VALUE	SCALE_X: Real NORM_X: SInt, Int, DInt, USInt, UInt, UDIInt, Real	Valore di ingresso da riportare in scala o normalizzare
MAX	SInt, Int, DInt, USInt, UInt, UDIInt, Real	Valore di ingresso massimo per il campo
OUT	SCALE_X: SInt, Int, DInt, USInt, UInt, UDIInt, Real NORM_X: Real	Valore di uscita riportato in scala o normalizzato

Nota

Il parametro VALUE di SCALE_X deve essere limitato a ($0,0 \leq \text{VALUE} \leq 1,0$)

Se il parametro VALUE è inferiore a 0,0 o superiore a 1,0:

- l'operazione di messa in scala lineare può generare valori OUT inferiori al valore del parametro MIN o superiori al valore del parametro MAX, che tuttavia rientrano nel campo consentito per il tipo di dati OUT. In questi casi l'esecuzione di SCALE_X imposta ENO = vero.
- È possibile generare valori in scala che non rientrano nel campo del tipo di dati OUT. In questi casi il parametro OUT viene impostato su un valore intermedio uguale alla parte meno significativa del numero reale riportato in scala prima della conversione finale nel tipo di dati OUT. In questo caso l'esecuzione di SCALE_X imposta ENO = falso.

Il parametro VALUE di NORM_X deve essere limitato a ($\text{MIN} \leq \text{VALUE} \leq \text{MAX}$)

Se il parametro VALUE è inferiore a MIN o superiore a MAX, l'operazione di messa in scala lineare può generare valori OUT normalizzati inferiori a 0,0 o superiori a 1,0. In questo caso l'esecuzione di NORM_X imposta ENO = vero.

Stato di ENO	Condizione	Risultato (OUT)
1	Nessun errore	Risultato valido
0	Il risultato è maggiore del campo valido per il tipo di dati OUT	Risultato intermedio: la parte meno significativa di un numero reale prima della conversione finale nel tipo di dati OUT.
0	Parametri MAX <= MIN	SCALE_X: la parte meno significativa del numero reale VALUE fino a raggiungere la dimensione di OUT. NORM_X: VALUE nel tipo di dati di VALUE ampliato fino a raggiungere le dimensioni di una doppia parola.
0	Parametro VALUE = +/- INF o +/- NaN	VALUE viene scritto in OUT

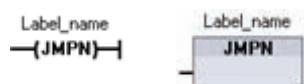
6.1.8 Controllo del programma

Istruzioni di salto e di etichetta

Le istruzioni di controllo del programma consentono di effettuare un controllo condizionato della sequenza di esecuzione:



JMP: se c'è flusso di corrente in ingresso alla bobina JMP (KOP) o se l'ingresso del box JMP è vero (FUP), l'esecuzione del programma prosegue con la prima istruzione successiva all'etichetta specificata.



JMPN: se non c'è flusso di corrente in ingresso alla bobina JMP (KOP) o se l'ingresso del box JMP è falso (FUP), l'esecuzione del programma prosegue con la prima istruzione successiva all'etichetta specificata.



Label: etichetta di destinazione per le istruzioni JMP e JMPN.

KOP **FUP**

Parametro	Tipo di dati	Descrizione
Label_name	Identificatore dell'etichetta	Identificatore per le istruzioni di salto e la corrispondente etichetta di destinazione del programma

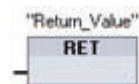
I nomi delle etichette possono essere digitati direttamente nell'istruzione LABEL. È possibile selezionare i nomi disponibili utilizzando l'icona di aiuto del parametro e inserirli nell'apposito campo delle istruzioni JMP e JMPN. Inoltre il nome dell'etichetta può essere digitato direttamente nell'istruzione JMP o JMPN.

Istruzione Return_Value (RET) di controllo dell'esecuzione

KOP



FUP



L'istruzione RET consente di concludere l'esecuzione del blocco attuale.

Parametro	Tipo di dati	Descrizione
Return_Value	Bool	Il parametro "Return_value" dell'istruzione RET viene assegnato all'uscita ENO del box di richiamo blocco contenuto nel blocco richiamante.

L'istruzione RET è opzionale e consente di concludere l'esecuzione del blocco attuale. Se è presente il flusso di corrente nella bobina RET (KOP) o se l'ingresso RET del box è vero (FUP), l'esecuzione del blocco attuale termina in quel punto e le istruzioni successive a RET non vengono eseguite. Se il blocco attuale è un OB, il parametro "Return_Value" viene ignorato. Se il blocco attuale è un FC o un FB, il valore del parametro "Return_Value" viene restituito alla routine richiamante come valore ENO del box richiamato.

Non è necessario inserire l'istruzione RET per ultima nel blocco perché questa operazione viene effettuata automaticamente. È possibile inserire più istruzioni RET nello stesso blocco.

Di seguito è illustrato un esempio di operazioni per l'utilizzo dell'istruzione RET in un blocco di codice FC:

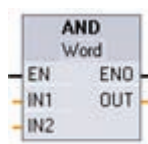
1. Creare un nuovo progetto e inserirvi un FC
2. Modificare l'FC:
 - Inserire istruzioni prelevandole dall'albero delle istruzioni.
 - Inserire un'istruzione RET, specificando quanto segue per il parametro "Return_Value":
Vero, falso o una locazione di memoria che specifichi il valore di ritorno richiesto.
 - Inserire altre istruzioni.
3. Richiamare l'FC da MAIN [OB1].

Per avviare l'esecuzione dell'FC l'ingresso EN del box FC nel blocco di codice MAIN deve essere vero.

Il valore specificato dall'istruzione RET nell'FC sarà presente nell'uscita ENO del box FC nel blocco di codice MAIN che segue l'FC per la quale è vero il flusso di corrente verso l'istruzione RET.

6.1.9 Operazioni logiche

Istruzioni AND, OR e XOR



AND: combinazione logica AND per i tipi di dati BYTE, WORD e DWORD

OR: combinazione logica OR per i tipi di dati BYTE, WORD e DWORD

XOR: combinazione logica OR esclusivo per i tipi di dati BYTE, WORD e DWORD

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Parametro	Tipo di dati	Descrizione
IN1, IN2	Byte, Word, DWord	Ingressi logici
OUT	Byte, Word, DWord	Uscita logica

I parametri IN1, IN2 e OUT vengono impostati tutti sul tipo di dati selezionato. I corrispondenti valori di bit di IN1 e IN2 vengono combinati logicamente per generare un risultato logico booleano nel parametro OUT. Dopo l'esecuzione di queste istruzioni ENO è sempre vero.

Istruzione Crea complemento a uno

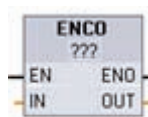


L'istruzione INV consente di calcolare il complemento a uno del parametro IN invertendo i valori dei singoli bit del parametro IN (modificando gli 0 in 1 e gli 1 in 0). Dopo l'esecuzione dell'istruzione ENO è sempre vero.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Parametro	Tipo di dati	Descrizione
IN	SInt, Int, DInt, USInt, UInt, UDIInt, Byte, Word, DWord	Elemento di dati da invertire
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Byte, Word, DWord	Uscita negata

Istruzioni Codifica e Decodifica



ENCO codifica un pattern di bit in un numero binario.

DENCO decodifica un numero binario in un pattern di bit.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Parametro	Tipo di dati	Descrizione
IN	ENCO: Byte, Word, DWord DECO: UInt	ENCO: pattern di bit da codificare DECO: valore da decodificare
OUT	ENCO: Int DECO: Byte, Word, DWord	ENCO: valore codificato DECO: pattern di bit decodificato

L'istruzione ENCO converte il parametro IN nel numero binario corrispondente alla posizione del bit impostato meno significativo del parametro IN e restituisce il risultato nel parametro OUT. Se il parametro IN è 0000 0001 o 0000 0000, viene restituito in OUT il valore 0. Se il valore del parametro IN è 0000 0000, ENO viene impostato su falso.

L'istruzione DECO decodifica il numero binario fornito dal parametro IN, impostando a 1 la corrispondente posizione di bit nel parametro OUT (gli altri bit vengono impostati a 0). Dopo l'esecuzione dell'istruzione DECO ENO è sempre vero.

Il tipo di dati del parametro OUT dell'istruzione DECO, che può essere Byte, Word, o DWord, limita il campo utile del parametro IN. Se il valore del parametro IN è maggiore del campo utile, viene eseguita un'operazione "modulo" per estrarre i bit meno significativi sotto indicati.

Campo del parametro IN di DECO:

- Vengono utilizzati 3 bit (valori 0-7) IN per impostare 1 posizione di bit in un byte OUT
- Vengono utilizzati 4 bit (valori 0-15) IN per impostare 1 posizione di bit in una parola OUT
- Vengono utilizzati 5 bit (valori 0-31) IN per impostare 1 posizione di bit in una doppia parola OUT

Valore IN di DECO		Valore OUT di DECO (decodifica di una posizione di bit)
		Byte OUT (8 bit):
IN min.	0	00000001
IN max.	7	10000000
		Word OUT (16 bit):
IN min.	0	0000000000000001
IN max.	15	1000000000000000
		DWord OUT: (32 bit):
IN min.	0	00000000000000000000000000000001
IN max.	31	10000000000000000000000000000000

Stato di ENO	Condizione	Risultato (OUT)
1	Nessun errore	Numero di bit valido
0	IN è zero	OUT viene impostata a zero

Istruzioni Seleziona (SEL) e Multiplexaggio (MUX)



- In funzione del valore assunto dal parametro G, SEL assegna al parametro OUT uno dei due valori di ingresso forniti.
- In funzione del valore assunto dal parametro K, MUX assegna al parametro OUT uno dei diversi valori di ingresso forniti. Se il valore del parametro K è maggiore del campo valido, viene assegnato al parametro OUT il valore del parametro ELSE.

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.



SEL	Tipo di dati	Descrizione
G	Bool	Selettore: • falso per IN0 • vero per IN1
IN0, IN1	SInt, Int, DInt, USInt, UInt, UDIInt, Real, Byte, Word, DWord, Time, Char	Ingressi
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, Byte, Word, DWord, Time, Char	Uscita

MUX	Tipo di dati	Descrizione
K	UInt	Valore del selettore: • 0 per IN0 • 1 per IN1 • ...
IN0, IN1,	SInt, Int, DInt, USInt, UInt, UDIInt, Real, Byte, Word, DWord, Time, Char	Ingressi
ELSE	SInt, Int, DInt, USInt, UInt, UDIInt, Real, Byte, Word, DWord, Time, Char	Valore di ingresso sostitutivo (opzionale)
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, Byte, Word, DWord, Time, Char	Uscita

Le variabili di ingresso e la variabile di uscita devono avere lo stesso tipo di dati.

- L'istruzione SEL seleziona sempre uno di due valori IN.
- L'istruzione MUX ha due parametri IN se inserita per prima nell'editor di programma, ma può essere ampliata con altri parametri IN.

Per aggiungere o eliminare i parametri di ingresso per l'istruzione MUX procedere nel seguente modo:

- Per aggiungere un ingresso fare clic con il tasto destro del mouse sul connettore dell'ingresso di uno dei parametri IN disponibili e selezionare il comando "Inserisci ingresso".
- Per eliminare un ingresso fare clic con il tasto destro del mouse sul connettore dell'ingresso di uno dei parametri IN disponibili e selezionare il comando "Cancella".

Codici delle condizioni di errore: Dopo l'esecuzione dell'istruzione SEL ENO è sempre vero.

Stato di ENO (MUX)	Condizione di MUX	Risultato di MUX (OUT)
1	Nessun errore	Il valore IN selezionato viene assegnato a OUT
0	K è maggiore di o uguale al numero di parametri IN	ELSE non fornito: OUT resta invariato ELSE fornito: il valore di ELSE viene assegnato a OUT

6.1.10 Scorrimento e rotazione

Istruzione di scorrimento



Le istruzioni di scorrimento consentono di far scorrere il pattern di bit del parametro IN. Il risultato viene assegnato al parametro OUT. Il parametro N specifica il numero di posizioni di bit fatte scorrere:

- SHR: fa scorrere un pattern di bit verso destra
- SHL: fa scorrere un pattern di bit verso sinistra

Fare clic sotto il nome del box e selezionare il tipo di dati nell'elenco a discesa.

Parametro	Tipo di dati	Descrizione
IN	Byte, Word, DWord	Pattern di bit da far scorrere
N	UInt	Numero di posizioni di bit da far scorrere
OUT	Byte, Word, DWord	Pattern di bit dopo l'operazione di scorrimento

- Se N=0 lo scorrimento non viene effettuato e il valore IN viene assegnato a OUT.
- Gli zeri vengono fatti scorrere nelle posizioni di bit liberate dall'operazione.
- Se il numero di posizioni da far scorrere (N) è maggiore di quello dei bit nel valore di destinazione (8 per Byte, 16 per Word, 32 per DWord), i valori di bit originali vengono fatti scorrere fuori e sostituiti con zeri (a OUT viene assegnato zero).
- Dopo l'esecuzione delle istruzioni di scorrimento ENO è sempre vero.

Esempio di SHL con dati Word: scorrimento di zeri da sinistra			
IN	1110 0010 1010 1101	Valore di OUT prima del primo scorrimento:	1110 0010 1010 1101
		Dopo il primo scorrimento verso sinistra:	1100 0101 0101 1010
		Dopo il secondo scorrimento verso sinistra:	1000 1010 1011 0100
		Dopo il terzo scorrimento verso sinistra:	0001 0101 0110 1000

Istruzione di rotazione



Le istruzioni di rotazione consentono di far ruotare il pattern di bit del parametro IN. Il risultato viene assegnato al parametro OUT. Il parametro N specifica il numero di posizioni di bit fatte ruotare:

- ROR: fa ruotare un pattern di bit verso destra
- ROL: fa ruotare un pattern di bit verso sinistra

Fare clic sotto il nome del box e selezionare il tipo di dati nel menu a discesa.

Parametro	Tipo di dati	Descrizione
IN	Byte, Word, DWord	Pattern di bit da far ruotare
N	UInt	Numero di posizioni di bit da far ruotare
OUT	Byte, Word, DWord	Pattern di bit dopo l'operazione di rotazione

- Se N=0 la rotazione non viene effettuata e il valore IN viene assegnato a OUT.
- I dati di bit fatti ruotare e uscire da un lato del valore di destinazione vengono reinseriti dal lato opposto, in modo da mantenere tutti i valori di bit originali.
- La rotazione viene eseguita anche se il numero di posizioni di bit da far ruotare (N) è superiore a quello del valore di destinazione (8 per Byte, 16 per Word, 32 per DWord).
- Dopo l'esecuzione delle istruzioni di rotazione ENO è sempre vero.

Esempio di ROR con dati WORD: rotazione dei bit che escono da destra e vengono reinseriti da sinistra			
IN	0100 0000 0000 0001	Valore di OUT prima della prima rotazione:	0100 0000 0000 0001
		Dopo la prima rotazione verso destra:	1010 0000 0000 0000
		Dopo la seconda rotazione verso destra:	0101 0000 0000 0000

6.2 Istruzioni avanzate

6.2.1 Parametri di errore comuni alle istruzioni avanzate

Le descrizioni delle istruzioni avanzate illustrano gli errori di runtime che possono verificarsi per ciascuna istruzione del programma. Oltre a questi errori si possono verificare gli errori comuni elencati di seguito. Se uno di questi errori si verifica durante l'esecuzione di un blocco di codice, la CPU passa in STOP a meno che non si inseriscano nel blocco le istruzioni GetError o GetErrorID per programmare una reazione all'errore.

Valore del codice della condizione (W#16#....)	Descrizione
8022	Area troppo piccola per l'ingresso
8023	Area troppo piccola per l'uscita
8024	Area di ingresso non ammessa
8025	Area di uscita non ammessa
8028	Assegnazione del bit di ingresso non ammessa
8029	Assegnazione del bit di uscita non ammessa
8030	L'area di uscita è un DB di sola lettura
803A	Il DB non esiste

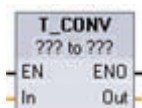
6.2.2 Istruzioni di orologio e calendario

Istruzioni di data e ora

Le istruzioni di data e ora consentono di programmare operazioni di calcolo della data e dell'ora.

- T_CONV converte il tipo di dati di un valore di data e ora: (da Time in DInt) o (da DInt in Time)
- T_ADD somma i valori di Time e DTL: (Time + Time = Time) o (DTL + Time = DTL)
- T_SUB sottrae i valori di Time e DTL: (Time - Time = Time) o (DTL - Time = DTL)
- T_DIFF fornisce la differenza tra i due valori DTL come valore Time: DTL - DTL = Time

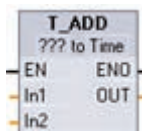
Tipo di dati	Dimensione (bit)	Campi validi
Time	32 Salvati come	Da T#-24d_20h_31m_23s_648ms a T#24d_20h_31m_23s_647ms Da -2.147.483.648 ms a +2.147.483.647 ms
Struttura dei dati DTL		
Anno: UInt	16	Da 1970 a 2554
Mese: UInt	8	Da 1 a 12
Giorno: UInt	8	Da 1 a 31
Giorno della settimana: UInt	8	Da 1=domenica a 7=sabato
Ora: UInt	8	Da 0 a 23
Minuto: UInt	8	Da 0 a 59
Secondo: UInt	8	Da 0 a 59
Nanosecondi: UInt	32	Da 0 a 999,999,999



T_CONV (conversione di data e ora) converte un tipo di dati Time in un tipo di dati DInt o all'inverso un tipo di dati DInt in un tipo di dati Time.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	DInt, Time	Immettere un valore Time o DInt
OUT	OUT	DInt, Time	Valore DInt o Time convertito

Selezionare i tipi di dati IN e OUT negli elenchi a discesa sotto il nome dell'istruzione.



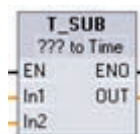
T_ADD (Somma data e ora) somma il valore dell'ingresso IN1 (tipi di dati DTL o Time) con quello dell'ingresso Time IN2. Il parametro OUT fornisce il risultato come valore DTL o Time.

È possibile eseguire due operazioni con i tipi di dati:

- Time + Time = Time
- DTL + Time = DTL

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN1	IN	DTL, Time	Valore DTL o Time
IN2	IN	Time	Valore Time da sommare
OUT	OUT	DTL, Time	Somma DTL o Time

Selezionare il tipo di dati IN1 nell'elenco a discesa sotto il nome dell'istruzione. Selezionando il tipo di dati IN1 viene impostato automaticamente il tipo di dati del parametro OUT.



T_SUB (Sottrai data e ora) sottrae il valore Time in IN2 da IN1 (valore DTL o Time). Il parametro OUT fornisce un valore differenziale con tipo di dati DTL o Time.

È possibile eseguire due operazioni con i tipi di dati:

- Time - Time = Time
- DTL - Time = DTL

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN1	IN	DTL, Time	Valore DTL o Time
IN2	IN	Time	Valore Time da sottrarre
OUT	OUT	DTL, Time	Differenza DTL o Time

Selezionare il tipo di dati IN1 nell'elenco a discesa sotto il nome dell'istruzione. Selezionando il tipo di dati IN1 viene impostato automaticamente il tipo di dati del parametro OUT.



T_DIFF (differenza di data e ora) sottrae il valore DTL in IN2 da DTL in IN1. Il parametro OUT fornisce un valore differenziale con tipo di dati Time.

- DTL - DTL = Time

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN1	IN	DTL	Valore DTL
IN2	IN	DTL	Valore DTL da sottrarre
OUT	OUT	Time	Differenza di tempo

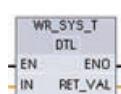
Codici delle condizioni di errore: ENO = 1 significa che non si è verificato alcun errore. ENO = 0 e parametro OUT = 0 errori:

- Valore DTL non valido
- Valore Time non valido

Istruzioni di orologio

Le istruzioni di orologio consentono di impostare e leggere l'orologio di sistema del PLC. Per i valori di data e ora viene utilizzato il tipo di dati DTL.

Struttura di DTL	Dimensione	Campi validi
Anno: UInt	16 bit	Da 1970 a 2554
Mese: UInt	8 bit	Da 1 a 12
Giorno: UInt	8 bit	Da 1 a 31
Giorno della settimana: UInt	8 bit	Da 1=domenica a 7=sabato
Ora: UInt	8 bit	Da 0 a 23
Minuto: UInt	8 bit	Da 0 a 59
Secondo: UInt	8 bit	Da 0 a 59
Nanosecondi: UInt	32 bit	Da 0 a 999,999,999



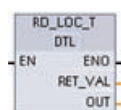
WR_SYS_T (Scrivi data e ora di sistema) imposta l'orologio hardware del PLC con un valore DTL nel parametro IN. Il valore non comprende gli offset per il fuso orario e l'ora legale.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	DTL	Ora da impostare nell'orologio di sistema del PLC
RET_VAL	OUT	Int	Codice della condizione di esecuzione



RD_SYS_T (Leggi data e ora di sistema) legge dal PLC la data e l'ora di sistema attuale. Questo valore non comprende gli offset per il fuso orario e l'ora legale.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
RET_VAL	OUT	Int	Codice della condizione di esecuzione
OUT	OUT	DTL	Ora di sistema attuale del PLC



RD_LOC_T (Leggi data e ora locale) fornisce la data e l'ora locali attuali del PLC indicandole con il tipo di dati DTL.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
RET_VAL	OUT	Int	Codice della condizione di esecuzione
OUT	OUT	DTL	Ora locale

- La data e l'ora locale vengono calcolate sulla base degli offset per il fuso orario e l'ora legale impostati dall'utente in Configurazione dispositivi, Orologio.
- La configurazione del fuso orario indica un offset rispetto al tempo coordinato universale (UTC).
- La configurazione dell'ora legale specifica il mese, la settimana, il giorno e l'ora di inizio dell'ora legale.
- Anche la configurazione dell'ora solare specifica il mese, la settimana, il giorno e l'ora di inizio dell'ora solare.
- L'offset del fuso orario viene applicato al valore dell'ora di sistema. L'offset dell'ora legale viene applicato solo quando è in vigore l'ora legale.

Codici delle condizioni di errore: ENO = 1 significa che non si è verificato alcun errore. ENO = 0 significa che si è verificato un errore di esecuzione e l'uscita RET_VAL fornisce il codice della relativa condizione.

RET_VAL (W#16#....)	Descrizione
0000	Nessun errore
8080	Ora locale non disponibile
8081	Valore dell'anno non valido
8082	Valore del mese non valido
8083	Valore del giorno non valido
8084	Valore dell'ora non valido
8085	Valore dei minuti non valido
8086	Valore dei secondi non valido
8087	Valore dei nanosecondi non valido
80B0	L'orologio hardware non funziona correttamente

6.2.3 Istruzioni con le stringhe e i caratteri

6.2.3.1 Descrizione dei dati String

Tipo di dati String

I dati String vengono salvati come intestazione di 2 byte seguita da max. 254 byte di caratteri ASCII. L'intestazione String contiene due lunghezze. Il primo byte corrisponde alla lunghezza massima indicata tra parentesi quadre durante l'inizializzazione della stringa oppure è impostato per default a 254. Il secondo byte dell'intestazione corrisponde alla lunghezza attuale ovvero al numero di caratteri validi della stringa. La lunghezza attuale deve essere inferiore o uguale alla lunghezza massima. Il numero di byte memorizzati per il formato String è superiore di 2 byte alla lunghezza massima.

Inizializzazione dei dati String

Per poter eseguire un'istruzione con le stringhe è innanzitutto necessario inizializzare i dati di ingresso e di uscita String come stringhe valide nella memoria.

Dati String validi

Una stringa valida deve avere una lunghezza massima superiore a zero e inferiore a 255. La lunghezza attuale deve essere inferiore o uguale alla lunghezza massima.

Le stringhe non possono essere assegnate alle aree di memoria I o Q.

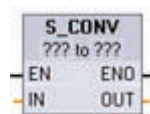
Per maggiori informazioni vedere: Formato del tipo di dati String (Pagina 65)

6.2.3.2 Istruzioni di conversione di stringhe

Conversioni di stringhe in valori e di valori in stringhe

Le seguenti istruzioni consentono di convertire stringhe di caratteri numerici in valori numerici o valori numerici in stringhe di caratteri numerici:

- S_CONV effettua una conversione (stringa numerica in valore numerico) o (valore numerico in stringa numerica)
- STRG_VAL converte una stringa numerica in valore numerico con opzioni per il formato
- VAL_STRG converte un valore numerico in una stringa numerica con opzioni per il formato



S_CONV (Converti stringa) converte una stringa di caratteri nel valore corrispondente o un valore nella corrispondente stringa di caratteri. L'istruzione S_CONV non dispone di funzioni di formattazione dell'uscita. È quindi più semplice ma meno flessibile delle istruzioni STRG_VAL e VAL_STRG.

Selezionare i tipi di dati dei parametri dagli elenchi a discesa.

S_CONV (conversioni da stringa in valore)

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String	Stringa di caratteri in ingresso
OUT	OUT	String, SInt, Int, DInt, USInt, UInt, UDInt, Real	Valore numerico in uscita

La conversione del parametro di stringa IN inizia dal primo carattere e continua fino alla fine della stringa o fino al primo carattere diverso da "0" ... "9", "+", "-", o ".". Il valore del risultato viene fornito nella posizione specificata nel parametro OUT. Se il valore numerico in uscita non rientra nel campo del tipo di dati OUT, il parametro OUT viene impostato a 0 e ENO viene impostato su falso. In caso contrario il parametro OUT contiene il risultato valido e ENO viene impostato su vero.

Regole per il formato della stringa in ingresso:

- Come separatore decimale della stringa IN si deve utilizzare il carattere ".".
- Le virgole "," come separatore delle migliaia a sinistra del separatore decimale sono consentite e ignorate.

- Gli spazi iniziali vengono ignorati.
- È consentita solo la rappresentazione in virgola fissa. I caratteri "e" ed "E" non vengono riconosciuti come notazione esponenziale.

S_CONV (conversioni da valore in stringa)

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String, SInt, Int, DInt, USInt, UInt, UDInt, Real	Valore numerico in ingresso
OUT	OUT	String	Stringa di caratteri in uscita

Un numero intero, un numero intero senza segno o un valore IN in virgola mobile vengono convertiti nella corrispondente stringa di caratteri in OUT. Perché la conversione sia possibile il parametro OUT deve far riferimento a una stringa valida. Una stringa valida è costituita dalla lunghezza massima della stringa nel primo byte, da quella attuale nel secondo byte e dai caratteri attuali della stringa nei byte successivi. La stringa convertita sostituisce i caratteri nella stringa OUT, a partire dal primo, e adegua il byte della lunghezza attuale della stringa OUT. Il byte della lunghezza massima della stringa OUT resta invariato.

Il numero di caratteri che vengono sostituiti varia in funzione del tipo di dati e del valore numerico del parametro IN. Il numero di caratteri sostituiti deve essere compreso entro la lunghezza di stringa del parametro OUT. La lunghezza massima (primo byte) della stringa OUT deve essere maggiore o uguale al numero massimo di caratteri convertiti previsto.

La seguente tabella mostra le lunghezze di stringa massime possibili necessarie per i diversi tipi di dati supportati.

Tipo di dati IN	Numero massimo di caratteri convertiti nella stringa OUT	Esempio	Lunghezza complessiva della stringa compresi i byte della lunghezza massima e di quella attuale
USInt	3	255	5
SInt	4	-128	6
UInt	5	65535	7
Int	6	-32768	8
UDInt	10	4294967295	12
DInt	11	-2147483648	13

Regole per il formato della stringa in uscita:

- I valori scritti nel parametro OUT non sono preceduti dal segno "+".
- Viene utilizzata la rappresentazione in virgola fissa (non la notazione esponenziale).
- Come separatore decimale per il parametro IN con tipo di dati Real viene utilizzato il punto ".".

Istruzione STRG_VAL



STRG_VAL (Converti stringa in valore) converte una stringa di caratteri numerici nel corrispondente numero intero o numero in virgola mobile. La conversione inizia nella stringa IN, a partire dall'offset di caratteri P, e continua fino alla fine della stringa o fino al primo carattere diverso da "+", "-", ".", ",", "e", "E" o "0" ... "9". Il risultato viene scritto nella posizione specificata nel parametro OUT.

Anche il parametro P viene restituito come offset nella stringa originale, inserito nel punto in cui è terminata la conversione. Per poter essere eseguiti come stringa valida nella memoria, i dati String devono essere inizializzati.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String	Stringa di caratteri ASCII da convertire
FORMAT	IN	Word	Opzioni per il formato di uscita
P	IN_OUT	UInt	IN: indice che punta al primo carattere da convertire (primo carattere = 1) OUT: indice che punta al primo carattere successivo alla fine della conversione
OUT	OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real	Valore numerico convertito

Parametro FORMAT di STRG_VAL

Qui di seguito è definito il parametro FORMAT dell'istruzione STRG_VAL. Le posizioni di bit inutilizzate devono essere impostate a zero.

Bit 16							Bit 8	Bit 7							Bit 0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	f r

f = formato di notazione 1 = notazione esponenziale
 0 = notazione in virgola fissa

r = formato del separatore 1 = "," (virgola)
 decimale 0 = "." (punto)

FORMAT (W#16#)	Formato di notazione	Separatore decimale
0000 (default)	Virgola fissa	"."
0001		","
0002	Esponenziale	"."
0003		","
Da 0004 a FFFF	Valori non ammessi	

Regole per la conversione di STRG_VAL:

- Se si utilizza il punto "." come separatore decimale, le virgole "," alla sua sinistra vengono interpretate come caratteri di separazione delle migliaia. Queste virgole sono ammesse e ignorate.
- Se si utilizza la virgola "," come separatore decimale, i punti "." alla sua sinistra vengono interpretati come caratteri di separazione delle migliaia. I punti sono ammessi e ignorati.
- Gli spazi iniziali vengono ignorati.

Istruzione VAL_STRG



VAL_STRG (Converti valore in stringa) converte un numero intero, un numero intero senza segno o un valore in virgola mobile nella corrispondente stringa di caratteri. Il valore rappresentato dal parametro IN viene convertito nella stringa indirizzata dal parametro OUT. Perché la conversione sia possibile il parametro OUT deve essere una stringa valida.

La stringa convertita sostituirà i caratteri della stringa OUT a partire dall'offset P fino al numero di caratteri specificato dal parametro SIZE. Il numero di caratteri indicato in SIZE deve essere compreso entro la lunghezza della stringa in OUT, a partire dalla posizione P. Questa istruzione è utile per inserire caratteri numerici nelle stringhe di testo. È ad esempio possibile immettere il numero "120" nella stringa "Pressione pompa = 120 psi".

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	SInt, Int, DInt, USInt, UInt, UDIInt, Real	Valore da convertire
SIZE	IN	USInt	Numero di caratteri da scrivere nella stringa OUT
PREC	IN	USInt	Precisione o dimensione della parte frazionaria. Il separatore decimale non è compreso.
FORMAT	IN	WORD	Opzioni per il formato di uscita
P	IN_OUT	UInt	IN: indice che punta al primo carattere di stringa OUT da sostituire (primo carattere = 1) OUT: indice che punta al primo carattere di stringa OUT successivo alla sostituzione
OUT	OUT	String	Stringa convertita

Il parametro PREC specifica la precisione o il numero di cifre della parte frazionaria della stringa. Se il valore del parametro IN è un numero intero, PREC specifica la posizione del separatore decimale. Se, ad esempio, il valore di dati è 123 e PREC = 1, il risultato sarà "12.3". La precisione massima supportata per il tipo di dati REAL è di 7 cifre.

Se il parametro P è maggiore della dimensione attuale della stringa OUT vengono aggiunti degli spazi fino alla posizione P e il risultato viene aggiunto alla fine della stringa. La conversione termina quando viene raggiunta la lunghezza di stringa massima in OUT.

Parametro FORMAT di VAL_STRG

Qui di seguito è definito il parametro FORMAT dell'istruzione VAL_STRG. Le posizioni di bit inutilizzate devono essere impostate a zero.

Bit 16							Bit 8	Bit 7								Bit 0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	s	f	r

s = segno del numero

1 = utilizzare il segno "+" e "-"

0 = utilizzare solo il segno "-"

f = formato di notazione

1 = notazione esponenziale

0 = notazione in virgola fissa

r = formato del separatore
decimale

1 = "," (virgola)

0 = "." (punto)

FORMATO (WORD)	Carattere del segno del numero	Formato di notazione	Separatore decimale
W#16#0000	Solo "-"	Virgola fissa	" "
W#16#0001			" "
W#16#0002		Esponenziale	" "
W#16#0003			" "
W#16#0004	"+" e "-"	Virgola fissa	" "
W#16#0005			" "
W#16#0006		Esponenziale	" "
W#16#0007			" "
Da W#16#0008 a W#16#FFFF	Valori non ammessi		

Regole per il formato della stringa nel parametro OUT:

- Se la stringa convertita non raggiunge la lunghezza specificata vi vengono aggiunti degli spazi introduttivi.
- Se il bit di segno del parametro FORMAT è falso, i valori di numero intero senza segno e con segno vengono scritti nel buffer di uscita senza il segno "+" iniziale. Il segno "-" viene utilizzato se necessario.
<spazi iniziali><cifre senza zeri iniziali>'.<cifre di PREC>
- Se il bit di segno è vero, i valori di numero intero senza segno e con segno vengono sempre scritti nel buffer di uscita preceduti dal segno.
<spazi iniziali><segno><cifre senza zeri iniziali>'.<cifre di PREC>
- Se FORMAT viene impostato su "notazione esponenziale", i valori con tipo di dati REAL vengono scritti nel buffer di uscita nel seguente modo:
<spazi iniziali><segno><cifra> '.' <cifre di PREC>'E' <segno><cifre senza zero iniziale>

- Se FORMAT viene impostato su "notazione in virgola fissa", i numeri interi, i numeri interi senza segno e i numeri con tipo di dati REAL vengono scritti nel buffer di uscita nel seguente modo:
<spazi iniziali><segno><cifre senza zeri iniziali>'.<cifre di PREC>
- Gli zeri iniziali a sinistra del separatore decimale (ad eccezione della cifra che lo segue direttamente) vengono eliminati.
- I valori a destra del separatore decimale vengono arrotondati in modo da rientrare nel numero di cifre a destra del separatore specificato nel parametro PREC.
- La dimensione della stringa deve essere di almeno tre byte superiore al numero di cifre a destra del separatore.
- I valori della stringa sono giustificati a destra.

Condizioni di errore rilevate da ENO

Se si verifica un errore durante la conversione vengono restituiti i seguenti risultati:

- ENO viene impostato a 0.
- OUT viene impostato a 0 o nel modo indicato negli esempi di conversione di una stringa in valore.
- OUT resta invariato o viene impostato nel modo indicato negli esempi in cui OUT è una stringa.

Stato di ENO	Descrizione
1	Nessun errore
0	Parametro non ammesso o non valido; ad esempio l'accesso a un DB che non esiste
0	Stringa non ammessa: la lunghezza massima della stringa sia 0 o 255
0	Stringa non ammessa: la lunghezza attuale è maggiore di quella massima
0	Il valore numerico convertito è troppo grande per il tipo di dati OUT specificato
0	La dimensione massima indicata nel parametro OUT deve essere sufficiente a contenere il numero di caratteri specificato dal parametro SIZE, a partire dalla posizione indicata dal parametro P
0	Valore P non ammesso: P=0 o P è maggiore della lunghezza attuale della stringa
0	Il parametro SIZE deve essere maggiore del parametro PREC

Esempi di conversione di stringa in valore con S_CONV

Stringa IN	Tipo di dati OUT	Valore OUT	ENO
"123"	Int/DInt	123	Vero
"-00456"	Int/DInt	-456	Vero
"123.45"	Int/DInt	123	Vero
" +2345"	Int/DInt	2345	Vero
"00123AB"	Int/DInt	123	Vero
"123"	Real	123.0	Vero
"123.45"	Real	123.45	Vero
"1.23e-4"	Real	1.23	Vero

Stringa IN	Tipo di dati OUT	Valore OUT	ENO
"1.23E-4"	Real	1.23	Vero
"12,345.67"	Real	12345.67	Vero
"3.4e39"	Real	3.4	Vero
"-3.4e39"	Real	-3.4	Vero
"1.17549e-38"	Real	1.17549	Vero
"12345"	SInt	0	Falso
"A123"	N/A	0	Falso
""	N/A	0	Falso
"++123"	N/A	0	Falso
"+-123"	N/A	0	Falso

Esempi di conversione di valore in stringa con S_CONV

Tipo di dati	Valore IN	Stringa OUT	ENO
UInt	123	"123"	Vero
UInt	0	"0"	Vero
UDInt	12345678	"12345678"	Vero
Real	-INF	"INF"	Falso
Real	+INF	"INF"	Falso
Real	NaN	"NaN"	Falso

Esempi di conversione con STRG_VAL

Stringa IN	FORMAT (W#16#....)	Tipo di dati OUT	Valore OUT	ENO
"123"	0000	Int/DInt	123	Vero
"-00456"	0000	Int/DInt	-456	Vero
"123.45"	0000	Int/DInt	123	Vero
"123.45"	0000	Int/DInt	2345	Vero
"00123AB"	0000	Int/DInt	123	Vero
"123"	0000	Real	123.0	Vero
"-00456"	0001	Real	-456.0	Vero
"123.45"	0001	Real	456.0	Vero
"123.45"	0000	Real	123.45	Vero
"123.45"	0001	Real	12345.0	Vero
"123,45"	0000	Real	12345.0	Vero
"123,45"	0001	Real	123.45	Vero
".00123AB"	0001	Real	123.0	Vero
"1.23e-4"	0000	Real	1.23	Vero
"1.23E-4"	0000	Real	1.23	Vero
"1.23E-4"	0002	Real	1.23E-4	Vero

Stringa IN	FORMAT (W#16#....)	Tipo di dati OUT	Valore OUT	ENO
"12,345.67"	0000	Real	12345.67	Vero
"12,345.67"	0001	Real	12.345	Vero
"3.4e39"	0002	Real	+INF	Vero
"-3.4e39"	0002	Real	-INF	Vero
"1.1754943e-38" (e inferiore)	0002	Real	0.0	Vero
"12345"	N/A	SInt	0	Falso
"A123"	N/A	N/A	0	Falso
""	N/A	N/A	0	Falso
"++123"	N/A	N/A	0	Falso
"-123"	N/A	N/A	0	Falso

Esempi di conversione con VAL_STRG

Gli esempi che seguono fanno riferimento a una stringa OUT inizializzata nel seguente modo:

"Current Temp = xxxxxxxxxxx C"

Il carattere "x" rappresenta gli spazi riservati al valore convertito.

Tipo di dati	Valore IN	P	SIZE	FORMAT (W#16#....)	PREC	Stringa OUT	ENO
UInt	123	16	10	0000	0	Current Temp = xxxxxxxx123 C	Vero
UInt	0	16	10	0000	2	Current Temp = xxxxxx0.00 C	Vero
UDInt	12345678	16	10	0000	3	Current Temp = x12345.678 C	Vero
UDInt	12345678	16	10	0001	3	Current Temp = x12345,678 C	Vero
Int	123	16	10	0004	0	Current Temp = xxxxxx+123 C	Vero
Int	-123	16	10	0004	0	Current Temp = xxxxxx-123 C	Vero
Real	-0.00123	16	10	0004	4	Current Temp = xxx-0.0012 C	Vero
Real	-0.00123	16	10	0006	4	Current Temp = -1.2300E-3 C	Vero
Real	-INF	16	10	N/A	4	Current Temp = xxxxxx-INF C	Falso
Real	+INF	16	10	N/A	4	Current Temp = xxxxxx+INF C	Falso
Real	NaN	16	10	N/A	4	Current Temp = xxxxxxxNaN C	Falso
UDInt	12345678	16	6	N/A	3	Current Temp = xxxxxxxxxxx C	Falso

6.2.3.3 Istruzioni con le stringhe

Il programma di comando può utilizzare le seguenti istruzioni con le stringhe e i caratteri per creare messaggi visualizzabili dall'operatore e log di processo.

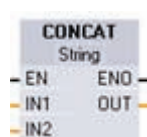
Errori comuni alle operazioni con le stringhe

Le istruzioni con le stringhe eseguite con le condizioni non ammesse o non valide sotto descritte danno come risultato ENO = 0 e una stringa di uscita nulla. Le condizioni di errore che si verificano per un'istruzione specifica sono elencate sotto la relativa descrizione.

ENO	Condizione	OUT
0	La lunghezza attuale di IN1 supera la lunghezza massima di IN1 oppure la lunghezza attuale di IN2 supera la lunghezza massima di IN2 (stringa non valida)	La lunghezza attuale viene impostata a 0
	La lunghezza massima di IN1, IN2 o OUT non è compresa entro il campo di memoria assegnato	
	La lunghezza massima di IN1, IN2 o OUT è 0 o 255 (lunghezza non ammessa)	



LEN: Leggi lunghezza stringa



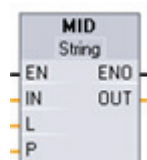
CONCAT: Concatena due stringhe



LEFT: Leggi sottostringa sinistra da stringa



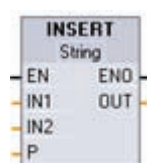
RIGHT: Leggi sottostringa destra da stringa



MID: Leggi sottostringa centrale da stringa



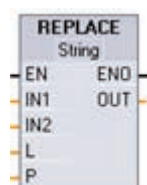
FIND: Trova sottostringa o carattere nella stringa



INSERT: Inserisci sottostringa nella stringa



DELETE: Cancella sottostringa da stringa



REPLACE: Sostituisci sottostringa nella stringa

Istruzione LEN

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String	Stringa di ingresso
OUT	OUT	UInt	Numero di caratteri validi della stringa IN

LEN (Lunghezza della stringa) indica la lunghezza attuale della stringa IN nell'uscita OUT. Le stringhe vuote hanno lunghezza zero. La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessuna condizione di stringa non valida	Lunghezza di stringa valida

Istruzione CONCAT

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN1	IN	String	Stringa di ingresso 1
IN2	IN	String	Stringa di ingresso 2
OUT	OUT	String	Stringa combinata (stringa 1 + stringa 2)

CONCAT (Concatena stringhe) unisce i parametri String IN1 e IN2 in modo da formare una stringa che verrà fornita in OUT. Dopo la concatenazione la stringa IN1 costituisce la parte sinistra e la stringa IN2 la parte destra della stringa combinata. La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Caratteri validi
0	La stringa risultante dalla concatenazione supera la lunghezza massima della stringa OUT	I caratteri della stringa risultante vengono copiati fino a raggiungere la lunghezza massima di OUT

Istruzione LEFT

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String	Stringa di ingresso
L	IN	Int	Lunghezza della sottostringa da creare utilizzando gli ultimi caratteri L a sinistra della stringa IN
OUT	OUT	String	Stringa di uscita

LEFT (Sottostringa sinistra) ricava una sottostringa costituita dai primi caratteri L del parametro di stringa IN.

- Se L è maggiore della lunghezza attuale della stringa IN, viene restituita in OUT l'intera stringa IN.
- Se la stringa in ingresso è vuota, viene restituita in OUT una stringa vuota.

La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Caratteri validi
0	L è inferiore o uguale a 0	La lunghezza attuale viene impostata a 0
	La lunghezza della sottostringa (L) da copiare è maggiore della lunghezza massima della stringa OUT	I caratteri vengono copiati fino a raggiungere la lunghezza massima di OUT

Istruzione RIGHT

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String	Stringa di ingresso
L	IN	Int	Lunghezza della sottostringa da creare utilizzando gli ultimi caratteri L a destra della stringa IN
OUT	OUT	String	Stringa di uscita

RIGHT (Sottostringa destra) ricava gli ultimi caratteri L di una stringa.

- Se L è maggiore della lunghezza attuale della stringa IN, viene restituita nel parametro OUT l'intera stringa IN.
- Se la stringa in ingresso è vuota, viene restituita in OUT una stringa vuota.

La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Caratteri validi
0	L è inferiore o uguale a 0	La lunghezza attuale viene impostata a 0
	La lunghezza della sottostringa (L) da copiare è maggiore della lunghezza massima della stringa OUT	I caratteri vengono copiati fino a raggiungere la lunghezza massima di OUT

Istruzione MID

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String	Stringa di ingresso
L	IN	Int	Lunghezza della sottostringa da creare utilizzando i caratteri L della stringa IN a partire dalla posizione P
P	IN	Int	Posizione del primo carattere della sottostringa da copiare: P= 1 indica la posizione iniziale dei caratteri della stringa IN
OUT	OUT	String	Stringa di uscita

MID (Sottostringa centrale) ricava la parte centrale di una stringa. La sottostringa centrale ha una lunghezza di L caratteri e inizia nella posizione P (compresa).

Se la somma di L e P supera la lunghezza attuale del parametro di stringa IN, viene restituita una sottostringa che inizia nella posizione P e continua fino alla fine della stringa IN. La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Caratteri validi
0	L o P è inferiore o uguale a 0	La lunghezza attuale viene impostata a 0
	P è maggiore della lunghezza massima di IN	
	La lunghezza della sottostringa (L) da copiare è maggiore della lunghezza massima della stringa OUT	I caratteri vengono copiati a partire dalla posizione P fino a raggiungere la lunghezza massima di OUT

Istruzione DELETE

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN	IN	String	Stringa di ingresso
L	IN	Int	Numero di caratteri da cancellare
P	IN	Int	Posizione del primo carattere da cancellare: il primo carattere della stringa IN occupa la posizione numero 1
OUT	OUT	String	Stringa di uscita

DELETE (Cancella sottostringa) cancella i caratteri L dalla stringa IN. La cancellazione dei caratteri inizia dalla posizione P (compresa) e nel parametro OUT viene restituita la sottostringa rimanente.

- Se L è uguale a zero la stringa in ingresso viene restituita in OUT.
- Se la somma di L e P è maggiore della lunghezza della stringa in ingresso, la stringa viene cancellata fino alla fine.

La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Caratteri validi
0	P è maggiore della lunghezza attuale di IN	IN viene copiato in OUT e non viene cancellato alcun carattere
	L è inferiore a 0 oppure P è inferiore o uguale a 0	La lunghezza attuale viene impostata a 0
	La stringa ottenuta dopo la cancellazione dei caratteri supera la lunghezza massima della stringa OUT	I caratteri della stringa risultante vengono copiati fino a raggiungere la lunghezza massima di OUT

INSERT

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN1	IN	String	Stringa di ingresso 1
IN2	IN	String	Stringa di ingresso 2
P	IN	Int	Ultima posizione dei caratteri nella stringa IN1 prima del punto di inserimento della stringa IN2. Il primo carattere della stringa IN1 occupa la posizione numero 1.
OUT	OUT	String	Stringa risultante

INSERT (Inserisci sottostringa) inserisce la stringa IN2 nella stringa IN1. L'inserimento inizia dopo il carattere nella posizione P. La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Caratteri validi
0	P è maggiore della lunghezza di IN1	IN2 viene concatenato a IN1 subito dopo l'ultimo carattere IN1
	P è inferiore o uguale a 0	La lunghezza attuale viene impostata a 0
	La stringa risultante dall'inserimento supera la lunghezza massima della stringa OUT	I caratteri della stringa risultante vengono copiati fino a raggiungere la lunghezza massima di OUT

REPLACE

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN1	IN	String	Stringa di ingresso
IN2	IN	String	Stringa di caratteri sostitutivi
L	IN	Int	Numero di caratteri da sostituire
P	IN	Int	Posizione del primo carattere da sostituire
OUT	OUT	String	Stringa risultante

REPLACE (Sostituisci sottostringa) sostituisce i caratteri L nel parametro di stringa IN1. La sostituzione inizia dalla posizione P (compresa) della stringa IN1 e i caratteri sostitutivi vengono forniti dal parametro di stringa IN2.

- Se il parametro L è uguale a zero, la stringa IN2 viene inserita nella posizione P della stringa IN1 senza che vengano cancellati caratteri dalla stringa IN1.
- Se P è uguale a 1, i primi caratteri L della stringa IN1 vengono sostituiti con i caratteri della stringa IN2.

La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Caratteri validi
0	P è maggiore della lunghezza di IN1	IN2 viene concatenato a IN1 subito dopo l'ultimo carattere IN1
	P punta all'interno di IN1, ma in IN1 rimane un numero di caratteri inferiore a quelli di L	IN2 sostituisce i caratteri finali di IN1 a partire dalla posizione P
	L è inferiore a 0 oppure P è inferiore o uguale a 0	La lunghezza attuale viene impostata a 0
	La stringa risultante dalla sostituzione supera la lunghezza massima della stringa OUT	I caratteri della stringa risultante vengono copiati fino a raggiungere la lunghezza massima di OUT

FIND

Parametro	Tipo di parametro	Tipo di dati	Descrizione
IN1	IN	String	Cerca nella stringa
IN2	IN	String	Cerca la stringa
OUT	OUT	Int	Posizione del primo elemento corrispondente ai criteri di ricerca all'interno della stringa IN1

FIND (Trova sottostringa) rileva la posizione della sottostringa o del carattere specificato da IN2 all'interno della stringa IN1. La ricerca inizia da sinistra. La posizione del primo elemento IN2 trovato nella stringa viene restituita in OUT. Se la stringa IN2 non viene trovata nella stringa IN1, viene restituito il valore zero. La seguente tabella elenca i codici delle condizioni per le istruzioni.

ENO	Condizione	OUT
1	Nessun errore rilevato	Posizione dei caratteri valida
0	IN2 è maggiore di IN1	La posizione dei caratteri viene impostata a 0

6.2.4 Istruzioni di comando del programma

6.2.4.1 Istruzione Riavvia watchdog del tempo di ciclo



RE_TRIGR (Riavvia watchdog del tempo di ciclo) consente di aumentare il tempo massimo che può trascorrere prima che il temporizzatore di controllo del tempo di ciclo generi un errore.

L'istruzione RE_TRIGR consente di riavviare il temporizzatore del tempo di ciclo durante un singolo ciclo. In questo modo il tempo di ciclo massimo consentito viene aumentato di un tempo di ciclo massimo a partire dall'ultima esecuzione della funzione RE_TRIGR.

La CPU limita l'uso dell'istruzione RE_TRIGR al ciclo del programma, ad esempio, all'OB1, e alle funzioni che questo richiama. Di conseguenza, se RE_TRIGR viene richiamata da uno qualsiasi degli OB riportati nell'elenco degli OB di ciclo, il temporizzatore di controllo del tempo di ciclo viene resettato e ENO = EN.

ENO = falso e il temporizzatore di controllo del tempo di ciclo non viene resettato se RE_TRIGR viene eseguita da un OB di avvio, un OB di allarme o un OB di errore.

Impostazione del tempo di ciclo massimo del PLC

Il valore del tempo del ciclo massimo può essere impostato con "Tempo di ciclo" nella finestra "Configurazione dispositivi" della CPU.

Controllo del tempo di ciclo	Valore minimo	Valore massimo	Valore di default
Tempo di ciclo massimo	1 ms	6000 ms	150 ms

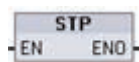
Timeout del watchdog

Se il temporizzatore di controllo del tempo di ciclo raggiunge il suo valore prima che sia terminato il ciclo di scansione, viene generato un errore. Se il programma utente contiene il blocco di gestione degli errori OB80 il PLC lo esegue nel punto in cui è possibile aggiungere della logica di programma per ottenere una reazione speciale. Se l'OB80 non è presente la prima condizione di timeout viene ignorata.

Se si verifica un secondo timeout del tempo di ciclo massimo durante lo stesso ciclo di programma (il doppio del valore di tempo di ciclo massimo), viene attivato un errore che commuta il PLC in STOP.

In STOP l'esecuzione del programma si arresta mentre la comunicazione e la diagnostica di sistema del PLC restano attive.

6.2.4.2 Istruzione Arresta ciclo di scansione del PLC



STP (Arresta ciclo di scansione del PLC) imposta il PLC in Stop. Quando il PLC è in Stop l'esecuzione del programma e gli aggiornamenti fisici dell'immagine di processo si arrestano.

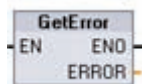
Per maggiori informazioni vedere: Configurazione delle uscite in caso di commutazione da RUN a STOP (Pagina 54)

Se EN = vero il PLC passa in STOP, l'esecuzione del programma si arresta e lo stato di ENO diventa non rilevante. Negli altri casi EN = ENO = 0.

6.2.4.3 Istruzioni di lettura degli errori

Le istruzioni di lettura degli errori forniscono informazioni sugli errori di esecuzione dei blocchi di programma. Se si inserisce un'istruzione GetError o GetErrorID nel blocco di codice è possibile gestire gli errori del programma all'interno del blocco.

GET_ERROR



GET_ERROR indica che si è verificato un errore di esecuzione di un blocco di programma e inserisce informazioni dettagliate sull'errore in una struttura di dati di errore predefinita.

Parametro	Tipo di dati	Descrizione
ERROR	ErrorStruct	Struttura dei dati dell'errore: È possibile modificare il nome della struttura ma non quello dei suoi membri.

Elemento di dati ErrorStruct	Tipo di dati	Descrizione
ERROR_ID	WORD	Identificatore dell'errore
FLAGS	BYTE	Sempre impostato a 0.
REACTION	BYTE	Reazione all'errore: • 0 = ignora; scrittura non effettuata (errore di scrittura) • 1 = valore sostitutivo: invece del valore di ingresso è stato utilizzato uno 0 (errore di lettura) • 2 = salta l'istruzione
BLOCK_TYPE	BYTE	Tipo di blocco in cui si è verificato l'errore: • 1 = OB • 2 = FC • 3 = FB
PAD_0	BYTE	Byte interno da definire a scopo di allineamento, sarà 0
CODE_BLOCK_NUMBER	UInt	Numero del blocco in cui si è verificato l'errore:
ADDRESS	UDInt	Indirizzo di memoria interna dell'istruzione in cui si è verificato l'errore
MODE	BYTE	Mappatura interna di come i campi rimanenti verranno interpretati per essere utilizzati da STEP 7 Basic
PAD_1	BYTE	Byte interno da definire a scopo di allineamento; non viene utilizzato, sarà 0
OPERAND_NUMBER	UInt	Numero interno dell'operando dell'istruzione
POINTER_NUMBER_LOCATION	UInt	(A) Indirizzo interno del puntatore dell'istruzione
SLOT_NUMBER_SCOPE	UInt	(B) Indirizzo di memoria interna
AREA	BYTE	(C) Area di memoria indirizzata quando si è verificato l'errore: • L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE • I: 16#81 • Q: 16#82 • M: 16#83 • DB: 16#84, 85, 8A, 8B
PAD_2	BYTE	Byte interno da definire a scopo di allineamento; non viene utilizzato, sarà 0

Elemento di dati ErrorStruct	Tipo di dati	Descrizione
DB_NUMBER	UInt	(D) DB indirizzato quando si è verificato un errore DB, 0 negli altri casi
OFFSET	UDInt	(E) Offset di bit indirizzato quando si è verificato l'errore (ad esempio: 12 = byte 1, bit 4)

GET_ERR_ID



GET_ERR_ID indica che si è verificato un errore di esecuzione di un blocco di programma e ne specifica l'ID (codice identificativo).

Parametro	Tipo di dati	Descrizione
ID	WORD	valori dell'ID dell'errore per il membro ERROR_ID di ErrorStruct

ERROR_ID esadecimale	ERROR_ID decimale	Errore di esecuzione del blocco di programma
2503	9475	Errore di puntatore non inizializzato
2522	9506	Errore di lettura di operando fuori campo
2523	9507	Errore di scrittura di operando fuori campo
2524	9508	Errore di lettura di un'area non valida
2525	9509	Errore di scrittura di un'area non valida
2528	9512	Errore di lettura dell'allineamento dei dati (allineamento errato dei bit)
2529	9513	Errore di scrittura dell'allineamento dei dati (allineamento errato dei bit)
2530	9520	DB protetto dalla scrittura
253A	9530	DB globale non presente
253C	9532	Versione errata o FC non presente
253D	9533	Istruzione non presente
253E	9534	Versione errata o FB non presente
253F	9535	Istruzione non presente
2575	9589	Errore di profondità di annidamento del programma
2576	9590	Errore di assegnazione dei dati locali
2942	10562	Ingresso fisico non presente
2943	10563	Uscita fisica non presente

Funzionamento

Per default la CPU reagisce all'errore di esecuzione di un blocco registrando un errore nel buffer di diagnostica e commutando in STOP. Se tuttavia si inseriscono una o più istruzioni GET_ERROR o ERR_ID all'interno di un blocco di codice, si fa in modo che questo gestisca gli errori al suo interno. In questo caso la CPU non commuta in STOP e non registra l'errore

nel buffer di diagnostica. Le informazioni di errore vengono invece riportate nell'uscita dell'istruzione GET_ERROR o GET_ERR_ID. Si può scegliere se leggere tutte le informazioni di errore con l'istruzione GET_ERROR o se leggere solo l'ID dell'errore con l'istruzione GET_ERR_ID. Generalmente il primo errore è quello più importante mentre quelli successivi sono una sua conseguenza.

La prima esecuzione di un'istruzione GET_ERROR o GET_ERR_ID all'interno di un blocco restituisce il primo errore rilevato durante l'esecuzione del blocco. L'errore può essersi verificato in qualsiasi momento tra l'inizio del blocco e l'esecuzione di GET_ERROR o GET_ERR_ID. Le esecuzioni successive di GET_ERROR o GET_ERR_ID restituiscono il primo errore successivo alla loro precedente esecuzione. La cronologia degli errori non viene salvata e l'esecuzione di un'istruzione riabilita il sistema PLC al rilevamento dell'errore successivo.

Il tipo di dati ErrorStruct utilizzato dall'istruzione GET_ERROR può essere aggiunto nell'editor di blocchi dati e negli editor di interfaccia dei blocchi in modo da consentire alla logica del programma di accedere ai valori di questo tipo. Per aggiungere questa struttura, selezionare ErrorStruct nell'elenco a discesa dei tipi di dati. È possibile creare più ErrorStruct definendoli con nomi univoci. I membri di un ErrorStruct non possono essere rinominati.

Condizione di errore indicata da ENO

Se EN = vero e viene eseguita GET_ERROR o GET_ERR_ID allora:

- ENO = vero indica che si è verificato un errore di esecuzione del blocco di codice e che sono presenti dati di errore
- ENO = falso indica che non si è verificato alcun errore di esecuzione del blocco di codice

È possibile collegare a ENO della logica di programma che reagisca all'errore; ENO si attiverà dopo che si è verificato un errore. Se è presente un errore il parametro di uscita ne salva i dati in un punto a cui il programma può accedere.

GET_ERROR e GET_ERR_ID possono essere utilizzate per trasmettere informazioni di errore dal blocco in esecuzione (blocco richiamato) al blocco richiamante. Inserire l'istruzione nell'ultimo segmento del blocco richiamato in modo che ne rilevi lo stato di esecuzione finale.

6.2.5 Istruzioni di comunicazione

6.2.5.1 Comunicazione Ethernet aperta

Comunicazione Ethernet aperta con collegamento/scollegamento automatico (TSEND_C e TRCV_C)

Nota

L'elaborazione di TSEND_C e TRCV_C può durare per un tempo indeterminato. Per fare in modo che queste istruzioni vengano elaborate in tutti i cicli di scansione le si deve richiamare sempre nel ciclo principale del programma, ad es. da un OB di ciclo o da un blocco di codice richiamato dal ciclo del programma. **Non** richiamare le istruzioni da un OB di interrupt di processo, un OB di allarme di ritardo, un OB di schedulazione orologio, un OB di allarme di errore o un OB di avvio.

Per informazioni sul trasferimento dei dati mediante queste istruzioni consultare il paragrafo sulla coerenza dei dati (Pagina 94).

Descrizione di TSEND_C

TSEND_C stabilisce un collegamento di comunicazione TCP o ISO on TCP con una stazione partner, trasmette i dati e può concludere il collegamento. Una volta impostato e stabilito, il collegamento viene mantenuto e controllato automaticamente dalla CPU. TSEND_C riunisce le funzioni di TCON, TDISCON e TSEND.

La dimensione minima dei dati trasmissibili con l'istruzione TSEND_C è di un byte.

Nota

L'impostazione di default del parametro LEN (LEN = 0) utilizza il parametro DATA per determinare la lunghezza dei dati trasmessi. I dati (DATA) trasmessi dall'istruzione TSEND_C devono avere le stesse dimensioni del parametro DATA dell'istruzione TRCV_C.

Le seguenti funzioni descrivono il funzionamento di TSEND_C:

- Per stabilire un collegamento eseguire TSEND_C con CONT = 1.
- Dopo che il collegamento è stato stabilito TSEND_C imposta il parametro DONE per un ciclo.
- Per concludere il collegamento eseguire TSEND_C con CONT = 0. Il collegamento viene interrotto immediatamente anche nella stazione ricevente. Il collegamento si interrompe in quel punto e i dati eventualmente presenti nel buffer di ricezione vengono cancellati.
- Per trasmettere i dati attraverso un collegamento attivo eseguire TSEND_C con un fronte di salita in REQ. Al termine della trasmissione, se non si verificano errori, TSEND_C imposta il parametro DONE per un ciclo.
- Per stabilire un collegamento e trasmettere i dati, eseguire TSEND_C con CONT = 1 e REQ = 1. Al termine della trasmissione TSEND_C imposta il parametro DONE per un ciclo.

Descrizione di TRCV_C

TRCV_C stabilisce un collegamento TCP o ISO on TCP con una CPU partner, riceve i dati e può concludere il collegamento. Una volta impostato e stabilito, il collegamento viene mantenuto e controllato automaticamente dalla CPU. L'istruzione TRCV_C riunisce le funzioni delle istruzioni TCON, TDISCON e TRCV.

La dimensione minima dei dati ricevibili con l'istruzione TRCV_C è di un byte. TRCV_C non supporta la trasmissione dei dati o degli array booleani.

Nota

L'impostazione di default del parametro LEN (LEN = 0) utilizza il parametro DATA per determinare la lunghezza dei dati trasmessi. I dati (DATA) trasmessi dall'istruzione TSEND_C devono avere le stesse dimensioni del parametro DATA dell'istruzione TRCV_C.

Le seguenti funzioni descrivono il funzionamento di TRCV_C:

- Per stabilire un collegamento eseguire TSEND_C con il parametro CONT = 1.
- Per ricevere i dati eseguire TRCV_C con il parametro EN_R = 1. TRCV_C riceve i dati ininterrottamente se il parametro EN_R = 1 e il parametro CONT = 1.
- Per concludere il collegamento eseguire TRCV_C con il parametro CONT = 0. Il collegamento viene annullato immediatamente e i dati eventualmente presenti potrebbero essere cancellati.

Modi di ricezione

TRCV_C gestisce gli stessi modi di ricezione dell'istruzione TRCV. La seguente tabella mostra come vengono immessi i dati nell'area di ricezione.

Tipo di protocollo	Immissione dei dati nell'area di ricezione	Parametro "connection_type"
TCP	Ricezione di dati con lunghezza specificata	B#16#11
ISO on TCP	Comandata da protocollo	B#16#12

Nota

Poiché TSEND viene elaborata in modo asincrono_C è necessario mantenere coerenti i dati dell'area di trasmissione finché il parametro DONE o ERROR non assume il valore "vero".

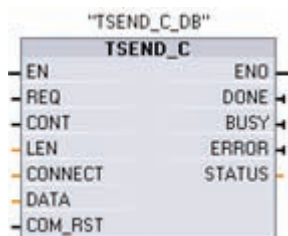
Nel caso di TSEND_C, lo stato "vero" nel parametro DONE significa che i dati sono stati trasmessi correttamente. Non significa che la CPU partner del collegamento ha effettivamente letto il buffer di ricezione.

Poiché TRCV_C viene elaborata in modo asincrono, i dati nell'area di ricezione sono coerenti solo se il parametro DONE = 1.

La seguente tabella mostra come interagiscono i parametri BUSY, DONE e ERROR.

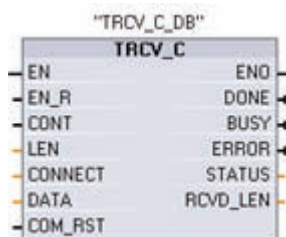
BUSY	DONE	ERROR	Descrizione
Vero	Non rilevante	Non rilevante	L'ordine è in corso di elaborazione.
Falso	Vero	Falso	L'ordine è stato concluso correttamente.
Falso	Falso	Vero	L'ordine si è concluso con un errore la cui causa è indicata nel parametro STATUS.
Falso	Falso	Falso	Non sono stati assegnati nuovi ordini.

Parametri di TSEND_C



Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	INPUT	Bool	In seguito a un fronte di salita il parametro di comando REQ avvia l'ordine di trasmissione attraverso il collegamento indicato in CONNECT.
CONT	INPUT	Bool	0: scollega 1: stabilisci e mantieni il collegamento
LEN	INPUT	Int	Numero massimo di byte da trasmettere. (default = 0, per cui il parametro DATA determina la lunghezza dei dati da trasmettere).
CONNECT	IN_OUT	TCON-Param	Puntatore alla descrizione del collegamento
DATI	IN_OUT	Versione	Area di trasmissione; contiene l'indirizzo e la lunghezza dei dati da trasmettere.
COM_RST	IN_OUT	Bool	1: riavvio del blocco funzionale completato, il collegamento esistente viene concluso.
DONE	OUTPUT	Bool	0: ordine non ancora avviato o ancora in corso. 1: ordine eseguito senza errori.
BUSY	OUTPUT	Bool	0: ordine concluso. 1: ordine non ancora concluso. Non è possibile riavviarne uno nuovo.
ERROR	OUTPUT	Bool	1: si è verificato un errore durante l'elaborazione. STATUS fornisce informazioni dettagliate sul tipo di errore.
STATUS	OUTPUT	WORD	informazioni di errore

Parametri di TRCV_C



Parametro	Tipo di parametro	Tipo di dati	Descrizione
EN_R	IN	Bool	Parametro di comando abilitato alla ricezione: se EN_R = 1, TRCV_C è pronta a ricevere. L'ordine di ricezione è in corso di elaborazione.
CONT	IN	Bool	Parametro di comando CONT: 0: scollega 1: stabilisci e mantieni il collegamento
LEN	IN	Int	Lunghezza dell'area di ricezione in byte. (default = 0, per cui il parametro DATA determina la lunghezza dei dati da trasmettere).
CONNECT	IN_OUT	TCON-Param	Puntatore alla descrizione del collegamento
DATI	IN_OUT	Versione	L'area di ricezione contiene l'indirizzo iniziale e la lunghezza massima dei dati ricevuti.
COM_RST	IN_OUT	Bool	1: concludi il riavvio del blocco funzionale, il collegamento esistente viene concluso.
DONE	OUT	Bool	0: ordine non ancora avviato o ancora in corso. 1: ordine eseguito senza errori.
BUSY	OUT	Bool	0: ordine concluso. 1: ordine non ancora concluso. Non è possibile riavviarne uno nuovo.
ERROR	OUT	Bool	1: si è verificato un errore durante l'elaborazione. STATUS fornisce informazioni dettagliate sul tipo di errore.
STATUS	OUT	WORD	informazioni di errore
RCVD_LEN	OUT	Int	Quantità di dati effettivamente ricevuti espressa in byte

Parametri Error e Status

ERROR	STATUS (W#16#...)	Descrizione
0	0000	ordine eseguito senza errori
0	7000	Nessun ordine in corso di elaborazione

ERROR	STATUS (W#16#...)	Descrizione
0	7001	Avvio dell'elaborazione dell'ordine, attivazione del collegamento, in attesa del partner del collegamento
0	7002	I dati vengono trasmessi o ricevuti
0	7003	Il collegamento viene concluso
0	7004	Collegamento stabilito e controllato, nessun ordine in corso di elaborazione
1	8085	Il parametro LEN è maggiore del valore massimo ammesso
1	8086	Il parametro CONNECT non è compreso entro il campo ammesso
1	8087	È stato raggiunto il numero massimo di collegamenti, non è possibile aggiungerne altri
1	8088	Il parametro LEN è più grande dell'area di memoria specificata in DATA, l'area di memoria di ricezione è troppo piccola
1	8089	Il parametro CONNECT non punta a un blocco dati.
1	8091	Profondità di annidamento massima superata
1	809A	Il parametro CONNECT punta a un campo che non rispetta la lunghezza della descrizione del collegamento.
1	809B	Il local_device_id nella descrizione del collegamento non corrisponde alla CPU.
1	80A1	Errore di comunicazione: - Il collegamento specificato non è ancora stato stabilito - Il collegamento specificato viene concluso e non può essere utilizzato per la trasmissione - L'interfaccia viene reinizializzata.
1	80A3	È in corso il tentativo di concludere un collegamento inesistente
1	80A4	L'indirizzo IP del collegamento con il partner remoto non è valido. A es., l'indirizzo IP del partner remoto è uguale all'indirizzo IP del partner locale.
1	80A7	Errore di comunicazione: TDISCON è stata richiamata prima che TCON terminasse (TDISCON deve prima concludere completamente il collegamento a cui fa riferimento l'ID)
1	80B2	Il parametro CONNECT punta a un blocco dati generato con la parola chiave UNLINKED
1	80B3	Parametri incoerenti: - Errore nella descrizione del collegamento - Porta locale (parametro local_tsap_id) è già presente in un'altra descrizione del collegamento - ID nella descrizione del collegamento diverso da quello specificato come parametro

ERROR	STATUS (W#16#...)	Descrizione
1	80B4	Utilizzando ISO on TCP (tipo_collegamento = B#16#12) per stabilire un collegamento passivo, il codice della condizione di errore 80B4 segnala che il TSAP immesso non è conforme a uno dei seguenti requisiti di indirizzo: - Se per il primo byte la lunghezza del TSAP locale è pari a 2 e il valore di ID del TSAP è pari a E0 o E1 (esadecimale), il secondo byte deve essere 00 o 01. - Se per il primo byte la lunghezza del TSAP locale è pari a 3 o maggiore e il valore di ID del TSAP è pari a E0 o E1 (esadecimale), il secondo byte deve essere 00 o 01 e tutti gli altri byte devono essere caratteri ASCII validi. - Se la lunghezza del TSAP locale è pari a 3 o maggiore e il primo byte dell'ID del TSAP non ha un valore pari a E0 o E1 (esadecimale), tutti gli altri byte dell'ID del TSAP devono essere caratteri ASCII validi. I caratteri ASCII validi sono valori di byte da 20 a 7E (esadecimale).
1	80C3	Tutte le risorse di collegamento sono state utilizzate.
1	80C4	Errore di comunicazione temporaneo: - non è possibile stabilire il collegamento ora - L'interfaccia sta ricevendo nuovi parametri - Il collegamento configurato viene eliminato da una TDISCON
1	8722	Parametro CONNECT: area di origine non valida: l'area non esiste nel DB
1	873A	Parametro CONNECT: l'accesso alla descrizione del collegamento non è possibile (ad es. DB non disponibile)
1	877F	Parametro CONNECT: errore interno, ad es. un riferimento ANY non valido

Comunicazione Ethernet aperta con controllo del collegamento/scollegamento

Nota

L'elaborazione delle istruzioni TCON, TDISCON, TSEND e TRCV può durare per un tempo indeterminato. Per fare in modo che queste istruzioni vengano elaborate in tutti i cicli di scansione le si deve richiamare sempre nel ciclo principale del programma, ad es. da un OB di ciclo o da un blocco di codice richiamato dal ciclo del programma. **Non** richiamare le istruzioni da un OB di interrupt di processo, un OB di allarme di ritardo, un OB di schedulazione orologio, un OB di allarme di errore o un OB di avvio.

Comunicaizone Ethernet mediante i protocolli TCP e ISO on TCP

Queste istruzioni del programma comandano il processo di comunicazione:

- TCON stabilisce un collegamento.
- TSEND e TRCV trasmettono e ricevono i dati.
- TDISCON interrompe il collegamento.

La dimensione minima dei dati trasmissibili o ricevibili con le istruzioni TSEND e TRCV è di un byte. TRCV non supporta la trasmissione dei dati o degli array booleani. Per informazioni sul trasferimento dei dati mediante queste istruzioni consultare il paragrafo sulla coerenza dei dati (Pagina 94).

Nota

L'impostazione di default del parametro LEN (LEN = 0) utilizza il parametro DATA per determinare la lunghezza dei dati trasmessi. I dati (DATA) trasmessi dall'istruzione TSEND devono avere le stesse dimensioni del parametro DATA dell'istruzione TRCV.

Entrambi i partner della comunicazione eseguono l'istruzione TCON per impostare e stabilire il collegamento. I parametri consentono di specificare i partner attivi e passivi del punto finale della comunicazione. Una volta impostato e stabilito, il collegamento viene mantenuto e controllato automaticamente dalla CPU.

Se il collegamento si conclude in seguito a un'interruzione della linea o a un'azione del partner di comunicazione remoto, ad esempio, il partner attivo cerca di ristabilirlo. Non è necessario eseguire nuovamente TCON.

Quando viene eseguita l'istruzione TDISCON o se la CPU è passata in STOP, il collegamento in corso viene concluso e quello impostato viene eliminato. Per impostare e ristabilire il collegamento è necessario eseguire nuovamente TCON.

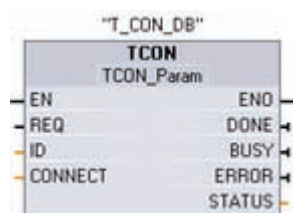
Descrizione della funzione

TCON, TDISCON, TSEND e TRCV funzionano in modo asincrono, per cui l'ordine viene elaborato nel corso di più esecuzioni dell'istruzione.

Si supponga, ad esempio, che si avvii un ordine per impostare e stabilire un collegamento eseguendo un'istruzione TCON con il parametro REQ = 1. Quindi si esegue TCON altre volte per controllare l'avanzamento dell'ordine e verificare che si concluda con il parametro DONE.

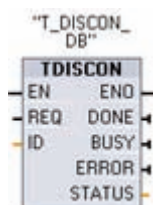
La seguente tabella mostra come interagiscono i parametri BUSY, DONE e ERROR. La tabella sotto riportata consente di determinare lo stato attuale dell'ordine.

BUSY	DONE	ERROR	Descrizione
Vero	Non rilevante	Non rilevante	L'ordine è in corso di elaborazione.
Falso	Vero	Falso	L'ordine è stato concluso correttamente.
Falso	Falso	Vero	L'ordine si è concluso con un errore la cui causa è indicata nel parametro STATUS.
Falso	Falso	Falso	Non sono stati assegnati nuovi ordini.

TCON

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Il parametro di comando REQUEST avvia l'ordine per stabilire il collegamento specificato dall'ID. L'ordine inizia in seguito a un fronte di salita.
ID	IN	CONN_OUC (Word)	Riferimento al collegamento da stabilire con il partner remoto o tra il programma utente e il livello di comunicazione del sistema operativo. L'ID deve essere identico all'ID di parametro associato, indicato nella descrizione del collegamento locale. Campo di valori: da W#16#0001 a W#16#0FFF
CONNECT	IN_OUT	TCON-Param	Puntatore alla descrizione del collegamento
DONE	OUT	Bool	Parametro di stato DONE: 0: ordine non ancora avviato o ancora in corso 1: ordine eseguito senza errori
BUSY	OUT	Bool	BUSY = 1: ordine non ancora concluso BUSY = 0: ordine concluso
ERROR	OUT	Bool	Parametro di stato ERROR: ERROR = 1: si è verificato un errore nell'elaborazione dell'ordine. STATUS fornisce informazioni dettagliate sul tipo di errore.
STATUS	OUT	WORD	Parametro di stato STATUS: informazioni di errore

TDISCON



TCP e ISO on TCP: TDISCON conclude un collegamento di comunicazione dalla CPU a un partner di comunicazione.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Il parametro di comando REQUEST avvia l'ordine per stabilire il collegamento specificato dall'ID. L'ordine inizia in seguito a un fronte di salita.
ID	IN	CONN_OUC (Word)	Riferimento al collegamento da concludere con il partner remoto o tra il programma utente e il livello di comunicazione del sistema operativo. L'ID deve essere identico all'ID di parametro associato, indicato nella descrizione del collegamento locale. Campo di valori: da W#16#0001 a W#16#0FFF

Parametro	Tipo di parametro	Tipo di dati	Descrizione
DONE	OUT	Bool	Parametro di stato DONE: 0: ordine non ancora avviato o ancora in corso 1: ordine eseguito senza errori
BUSY	OUT	Bool	BUSY = 1: ordine non ancora concluso BUSY = 0: ordine concluso
ERROR	OUT	Bool	ERROR = 1: si è verificato un errore durante l'elaborazione.
STATUS	OUT	WORD	Codice di errore

TSEND



Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Il parametro di comando REQUEST avvia l'ordine di trasmissione in seguito a un fronte di salita. I dati vengono trasferiti dall'area specificata da DATA e LEN.
ID	IN	CONN_OUC (Word)	Riferimento al collegamento associato. L'ID deve essere identico all'ID di parametro associato, indicato nella descrizione del collegamento locale. Campo di valori: da W#16#0001 a W#16#0FFF
LEN	IN	Int	Numero massimo di byte da trasmettere con l'ordine
DATA	IN_OUT	Versione	Puntatore all'area di dati da trasmettere: Area di trasmissione; contiene l'indirizzo e la lunghezza. L'indirizzo fa riferimento a quanto segue: - Tabella dell'immagine di processo degli ingressi - Tabella dell'immagine di processo delle uscite - Un merker - Un blocco dati
DONE	OUT	Bool	Parametro di stato DONE: 0: ordine non ancora avviato o ancora in corso. 1: ordine eseguito senza errori.
BUSY	OUT	Bool	- BUSY = 1: ordine non ancora concluso. Non è possibile riavviarne uno nuovo. - BUSY = 0: ordine concluso.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
ERROR	OUT	Bool	Parametro di stato ERROR: ERROR = 1: si è verificato un errore durante l'elaborazione. STATUS fornisce informazioni dettagliate sul tipo di errore
STATUS	OUT	WORD	Parametro di stato STATUS: informazioni di errore

TRCV



Parametro	Tipo di parametro	Tipo di dati	Descrizione
EN_R	IN	Bool	Parametro di comando abilitato alla ricezione: se EN_R = 1, TRCV è pronta a ricevere. L'ordine di ricezione è in corso di elaborazione.
ID	IN	CONN_OUC (Word)	Riferimento al collegamento associato. L'ID deve essere identico all'ID di parametro associato, indicato nella descrizione del collegamento locale. Campo di valori: da W#16#0001 a W#16#0FFF
LEN	IN	Int	Lunghezza dell'area di ricezione in byte (default = 0, per cui il parametro DATA determina la lunghezza dei dati da trasmettere).
DATI	IN_OUT	Versione	Puntatore ai dati ricevuti: Area di ricezione che contiene l'indirizzo e la lunghezza. L'indirizzo fa riferimento a quanto segue: • Tabella dell'immagine di processo degli ingressi • Tabella dell'immagine di processo delle uscite • Un merker • Un blocco dati
NDR	OUT	Bool	Parametro di stato NDR: • NDR = 0: ordine non ancora avviato o ancora in corso. • NDR = 1: ordine concluso correttamente.
BUSY	OUT	Bool	• BUSY = 1: ordine non ancora concluso. Non è possibile riavviarne uno nuovo. • BUSY = 0: ordine concluso.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
ERROR	OUT	Bool	ERROR=1: si è verificato un errore durante l'elaborazione. STATUS fornisce informazioni dettagliate sul tipo di errore.
STATUS	OUT	WORD	informazioni di errore
RCVD_LEN	OUT	Int	Quantità di dati effettivamente ricevuti espressa in byte

Area di ricezione

L'istruzione TRCV scrive i dati ricevuti in un'area di ricezione specificata dalle due seguenti variabili:

- Puntatore all'inizio dell'area
- Lunghezza dell'area

Nota

L'impostazione di default del parametro LEN (LEN = 0) utilizza il parametro DATA per determinare la lunghezza dei dati trasmessi. I dati (DATA) trasmessi dall'istruzione TSEND devono avere le stesse dimensioni del parametro DATA dell'istruzione TRCV.

La seguente tabella mostra in che modo TRCV scrive i dati ricevuti nell'area di ricezione.

Tipo di protocollo	Immissione dei dati nell'area di ricezione	Parametro "connection_type"
TCP	Ricezione di dati con lunghezza specificata	B#16#11
ISO on TCP	Comandata da protocollo	B#16#12

Non appena vengono ricevuti tutti i dati dell'ordine, TRCV li trasferisce nell'area di ricezione e imposta NDR a 1.

Codici delle condizioni di errore per TCON

ERROR	STATUS (W#16#...)	Spiegazione
0	0000	Il collegamento è stato stabilito correttamente
0	7000	Nessun ordine in corso di elaborazione
0	7001	Avvio dell'elaborazione dell'ordine, attivazione del collegamento
0	7002	Chiamata successiva automatica (REQ non rilevante), il collegamento viene stabilito
1	8086	Il parametro ID non è compreso entro il campo ammesso.
1	8087	È stato raggiunto il numero massimo di collegamenti, non è possibile aggiungerne altri
1	809B	Il local_device_id nella descrizione del collegamento non corrisponde alla CPU.
1	80A1	Il collegamento o la porta sono già occupati dall'utente

ERROR	STATUS (W#16#...)	Spiegazione
1	80A2	La porta locale o remota è occupata dal sistema
1	80A3	È in corso il tentativo di ristabilire un collegamento esistente
1	80A4	L'indirizzo IP del punto finale del collegamento remoto non è valido; potrebbe corrispondere all'indirizzo IP locale
1	80A7	Errore di comunicazione: TDISCON è stata eseguita prima della fine di TCON. TDISCON deve prima concludere completamente il collegamento a cui fa riferimento l'ID.
1	80B3	Assegnazione di parametro incoerente: errore cumulativo per i codici di errore W#16#80A0 - W#16#80A2, W#16#80A4, W#16#80B4 - W#16#80B9
1	80B4	Utilizzando ISO on TCP (tipo_collegamento = B#16#12) per stabilire un collegamento passivo, il codice della condizione di errore 80B4 segnala che il TSAP immesso non è conforme a uno dei seguenti requisiti di indirizzamento: - Se per il primo byte la lunghezza del TSAP locale è pari a 2 e il valore di ID del TSAP è pari a E0 o E1 (esadecimale), il secondo byte deve essere 00 o 01. - Se per il primo byte la lunghezza del TSAP locale è pari o superiore a 3 e il valore di ID del TSAP è pari a E0 o E1 (esadecimale), il secondo byte deve essere 00 o 01 e tutti gli altri byte devono essere caratteri ASCII validi. - Se la lunghezza del TSAP locale è pari o superiore a 3 e il primo byte dell'ID del TSAP non ha un valore pari a E0 o E1 (esadecimale), tutti gli altri byte dell'ID del TSAP devono essere caratteri ASCII validi. I caratteri ASCII validi sono valori di byte da 20 a 7E (esadecimale).
1	80B5	Errore nel parametro active_est
1	80B6	Errore di assegnazione nel parametro connection_type
1	80B7	Errore in uno dei seguenti parametri: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len
1	80B8	Il parametro nella descrizione del collegamento locale e l'ID del parametro sono diversi
1	80C3	Tutte le risorse di collegamento sono state utilizzate.
1	80C4	Errore di comunicazione temporaneo: - non è possibile stabilire il collegamento ora. - L'interfaccia sta ricevendo nuovi parametri. - Il collegamento configurato viene eliminato da TDISCON.

Codici delle condizioni di errore per TDISCON

ERROR	STATUS (W#16#...)	Spiegazione
0	0000	Il collegamento è stato concluso correttamente
0	7000	Nessun ordine in corso di elaborazione
0	7001	Avvio dell'elaborazione dell'ordine, il collegamento viene concluso
0	7002	Chiamata successiva automatica (REQ non rilevante), il collegamento viene concluso
1	8086	Il parametro ID non è compreso entro il campo ammesso per gli indirizzi.

ERROR	STATUS (W#16#...)	Spiegazione
1	80A3	È in corso il tentativo di concludere un collegamento inesistente
1	80C4	Errore di comunicazione temporaneo: L'interfaccia sta ricevendo nuovi parametri o il collegamento viene stabilito.

Codici delle condizioni di errore per TSEND

ERROR	STATUS (W#16#...)	Spiegazione
0	0000	Ordine di trasmissione concluso senza errori
0	7000	Nessun ordine in corso di elaborazione
0	7001	Avvio dell'elaborazione dell'ordine, i dati vengono trasmessi: durante l'elaborazione il sistema operativo accede ai dati nell'area di trasmissione DATA.
0	7002	Chiamata successiva automatica (REQ non rilevante), l'ordine è in corso di elaborazione: durante l'elaborazione il sistema operativo accede ai dati nell'area di trasmissione DATA.
1	8085	Il parametro LEN è maggiore del valore massimo ammesso.
1	8086	Il parametro ID non è compreso entro il campo ammesso per gli indirizzi
1	8088	Il parametro LEN è maggiore dell'area di memoria specificata in DATA
1	80A1	Errore di comunicazione: - Il collegamento specificato non è ancora stato stabilito - Il collegamento specificato viene concluso e non può essere utilizzato per la trasmissione. - L'interfaccia viene reinizializzata.
1	80C3	Mancanza interna di risorse: un blocco con questo ID è già in corso di elaborazione in una diversa classe di priorità
1	80C4	Errore di comunicazione temporaneo: - Non è possibile stabilire ora il collegamento con il partner di comunicazione. - L'interfaccia sta ricevendo nuovi parametri o il collegamento viene stabilito.

Codici delle condizioni di errore per TRCV

ERROR	STATUS (W#16#...)	Spiegazione
0	0000	Nuovi dati accettati: la lunghezza attuale dei dati ricevuti viene indicata in RCVD_LEN.
0	7000	Blocco non pronto a ricevere
0	7001	Blocco pronto a ricevere, l'ordine di ricezione è stato attivato.
0	7002	Chiamata successiva automatica, l'ordine di ricezione è in corso di elaborazione: durante l'elaborazione i dati vengono scritti nell'area di ricezione. Un errore potrebbe quindi determinare dati incoerenti in tale area.

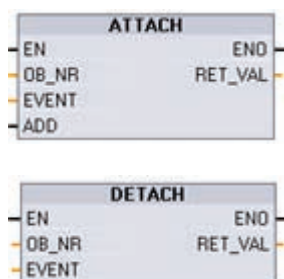
ERROR	STATUS (W#16#...)	Spiegazione
1	8085	Il parametro LEN è maggiore del valore massimo ammesso oppure dopo la prima chiamata è stato modificato il parametro LEN o DATA.
1	8086	Il parametro ID non è compreso entro il campo ammesso per gli indirizzi
1	8088	L'area di ricezione è troppo piccola: il valore LEN è maggiore dell'area di ricezione specificata da DATA.
1	80A1	Errore di comunicazione: - Il collegamento specificato non è ancora stato stabilito - Il collegamento specificato viene concluso e non può essere utilizzato per eseguire un ordine di ricezione. - L'interfaccia sta ricevendo nuovi parametri.
1	80C3	Mancanza interna di risorse: un blocco con questo ID è già in corso di elaborazione in una diversa classe di priorità
1	80C4	Errore di comunicazione temporaneo: - Non è possibile stabilire ora il collegamento con il partner di comunicazione. - L'interfaccia sta ricevendo nuove impostazioni di parametri oppure il collegamento viene stabilito.

6.2.5.2 Istruzioni punto a punto (PtP)

Il paragrafo Comunicazione punto a punto (PtP) (Pagina 251) fornisce informazioni dettagliate sulle istruzioni PtP e i moduli di comunicazione.

6.2.6 Istruzioni di allarme

6.2.6.1 Istruzioni di assegnazione e separazione



Le istruzioni ATTACH e DETACH consentono di attivare e disattivare sottoprogrammi comandati da eventi.

- ATTACH attiva l'esecuzione del sottoprogramma di un OB di allarme per un dato evento di interrupt di processo.
- DETACH disattiva l'esecuzione del sottoprogramma di un OB di allarme per un dato evento di interrupt di processo.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
OB_NR	IN	Int	Identificativo del blocco organizzativo: scegliere tra gli OB di interrupt di processo disponibili creati con la funzione "Inserisci nuovo blocco". Fare doppio clic sul campo del parametro, quindi fare clic sull'icona di aiuto per visualizzare gli OB disponibili.
EVENT	IN	DWord	Identificativo dell'evento: scegliere tra gli eventi di interrupt di processo disponibili che sono stati attivati in Configurazione dispositivi della CPU per gli ingressi digitali o i contatori veloci. Fare doppio clic sul campo del parametro, quindi fare clic sull'icona di aiuto per visualizzare gli eventi disponibili.
ADD (solo ATTACH)	IN	Bool	ADD = 0 (default): questo evento sostituisce tutte le precedenti assegnazioni di eventi effettuate per questo OB. ADD = 1: questo evento viene aggiunto alle precedenti assegnazioni di eventi effettuate per questo OB.
RET_VAL	OUT	Int	Codice della condizione di esecuzione

Eventi di interrupt di processo

La CPU supporta i seguenti eventi di interrupt di processo:

- Eventi "fronte di salita" (tutti gli ingressi digitali integrati nella CPU più qualsiasi ingresso digitale della signal board)
 - Quando l'ingresso digitale passa da OFF a ON in risposta ad una variazione del segnale proveniente dall'apparecchiatura da campo a cui è collegato si verifica un fronte di salita.
- Eventi "fronte di discesa" (tutti gli ingressi digitali integrati nella CPU più qualsiasi ingresso della signal board)
 - Quando l'ingresso digitale passa da ON a OFF si verifica un fronte di discesa.
- Eventi "valore attuale del contatore veloce (HSC) = valore di riferimento (CV = RV)" (HSC da 1 a 6)
 - Quando il valore attuale di conteggio passa da un valore adiacente al valore che coincide esattamente con quello di riferimento precedentemente definito viene generato un allarme CV = RV per un HSC.
- Eventi "cambiamento di direzione HSC" (HSC da 1 a 6)
 - Quando il sistema rileva che il conteggio dell'HSC cambia da crescente in decrescente o viceversa si verifica un evento "cambiamento di direzione".
- Eventi "reset esterno HSC" (HSC da 1 a 6)
 - Alcuni modi degli HSC consentono di assegnare un ingresso digitale come reset esterno che ha la funzione di azzerare il conteggio dell'HSC. Quando questo ingresso passa da OFF a ON si verifica un evento di reset esterno per l'HSC.

Attivazione degli eventi di interrupt di processo in Configurazione dispositivi

Durante la configurazione dei dispositivi è necessario attivare gli interrupt di processo. Se si desidera assegnare questo evento durante la configurazione o il runtime è quindi necessario selezionare, per un canale di ingresso o un HSC, la casella di attivazione degli eventi in Configurazione dispositivi.

Opzioni disponibili in Configurazione dispositivi del PLC:

- Ingresso digitale
 - Attiva rilevazione del fronte di salita
 - Attiva rilevazione del fronte di discesa
- Contatore veloce (HSC)
 - Attiva questo contatore veloce
 - Genera allarme per evento con valore di conteggio uguale al valore di riferimento
 - Genera allarme per evento di resettaggio esterno
 - Genera allarme per evento di cambio direzione

Inserimento di nuovi OB di interrupt di processo nel programma

Per default gli eventi non sono associati ad alcun OB prima di essere attivati per la prima volta. Questa condizione è segnalata dall'etichetta "<non collegato>" nell'elenco a discesa "Interrupt di processo:" della Configurazione dispositivi. Ad un evento di interrupt di processo possono essere assegnati solamente OB di interrupt di processo. Tutti gli OB di interrupt di processo compaiono nell'elenco a discesa "Interrupt di processo:". Se l'elenco non contiene alcun OB, creare un OB di tipo "Interrupt di processo" procedendo come indicato di seguito. Nella diramazione "Blocchi di programma" dell'albero del progetto:

1. fare doppio clic su "Inserisci nuovo blocco", selezionare "Blocco organizzativo (OB)" e quindi "Interrupt di processo".
2. Se si desidera rinominare l'OB, selezionare il linguaggio di programmazione (KOP o FUP) e quindi il numero del blocco (passare alla modalità manuale e scegliere un numero di blocco diverso da quello proposto).
3. Modificare l'OB e indicare la reazione che il programma deve attivare quando si verifica l'evento. Da questo OB è possibile richiamare FC e FB fino a un massimo di quattro livelli di annidamento.

Parametro OB_NR

Tutti i nomi degli OB di interrupt di processo esistenti sono riportati nell'elenco a discesa "Interrupt di processo:" di Configurazione dispositivi e nell'elenco a discesa del parametro OB_NR nei box delle istruzioni ATTACH / DETACH.

Parametro EVENT

Ad ogni evento di interrupt di processo attivato è assegnato un nome di default univoco. Il nome può essere modificato nella casella "Nome evento:", ma deve essere in ogni caso univoco. Questi nomi vengono utilizzati nella tabella delle variabili "Costanti" e compariranno nell'elenco a discesa del parametro EVENT per i box delle istruzioni ATTACH e DETACH. Il valore della variabile è un numero interno utilizzato per identificare l'evento.

Funzionamento generale

Ogni evento di processo può essere assegnato a un OB di interrupt di processo. Questo verrà inserito in una coda d'attesa ed eseguito quando si verifica l'evento previsto. L'assegnazione dell'evento all'OB può essere effettuata durante la configurazione o il runtime.

In fase di configurazione l'utente può scegliere se assegnare o separare un OB da un evento attivato. Per assegnare un OB ad un evento durante la configurazione, utilizzare l'elenco a discesa "Interrupt di processo:" (fare clic sulla freccia verso il basso sulla destra) e selezionare un OB di interrupt di processo tra quelli disponibili. Selezionare il nome dell'OB dall'elenco oppure scegliere "<non collegato>" per eliminare l'assegnazione.

Gli eventi di interrupt di processo attivati possono essere assegnati o separati anche durante il runtime, utilizzando le istruzioni di programma ATTACH o DETACH (se necessario anche più volte). Se non è stato assegnato alcun OB (perché è stato selezionato "<non collegato>" in Configurazione dispositivi oppure è stata eseguita un'istruzione DETACH), l'evento di allarme di processo attivo viene ignorato.

Utilizzo di DETACH

L'istruzione DETACH consente di separare un particolare evento o tutti gli eventi da un determinato OB. Se è stato specificato un particolare evento (EVENT), questo sarà l'unico ad essere separato dall'OB_NR indicato, mentre gli altri eventi assegnati allo stesso OB rimarranno invariati. Se invece non si specifica alcun evento particolare verranno separati tutti gli eventi assegnati a quell'OB (OB_NR).

Codici delle condizioni di errore

RET_VAL (W#16#....)	Stato di ENO	Descrizione
0000	1	Nessun errore
0001	0	Nessun evento da separare (solo DETACH)
8090	0	OB non presente
8091	0	Tipo dell'OB errato
8093	0	Evento non presente

6.2.6.2 Istruzioni Avvia allarme di ritardo e Annulla allarme di ritardo

Le istruzioni SRT_DINT e CAN_DINT consentono di avviare e annullare l'elaborazione degli allarmi di ritardo. Ogni allarme di ritardo è un evento che si verifica una sola volta allo scadere di un tempo specificato. Se l'evento di ritardo viene annullato prima dello scadere del tempo previsto, l'allarme non si verifica.



SRT_DINT avvia un allarme di ritardo che esegue il sottoprogramma di un OB (blocco organizzativo) allo scadere del ritardo specificato dal parametro DTIME.



CAN_DINT annulla un allarme di ritardo già avviato. In questo caso l'OB di allarme di ritardo non viene eseguito.

Parametri SRT_DINT

Parametro	Tipo di parametro	Tipo di dati	Descrizione
OB_NR	IN	Int	Blocco organizzativo (OB) da avviare allo scadere di un ritardo: scegliere tra gli OB di allarme di ritardo disponibili che sono stati creati utilizzando la funzione dell'albero del progetto "Inserisci nuovo blocco". Fare doppio clic sul campo del parametro, quindi fare clic sull'icona di aiuto per visualizzare gli OB disponibili.
DTIME	IN	Time	Valore del ritardo (da 1 a 60000 ms) È anche possibile ottenere tempi di ritardo più lunghi, ad esempio inserendo un contatore in un OB di allarme di ritardo.
SIGN	IN	WORD	Non utilizzato nell'S7-1200; è ammesso qualsiasi valore
RET_VAL	OUT	Int	Codice della condizione di esecuzione

Parametri CAN_DINT

Parametro	Tipo di parametro	Tipo di dati	Descrizione
OB_NR	IN	Int	Identificativo per l'OB di allarme di ritardo. È possibile utilizzare un numero di OB o un nome simbolico.
RET_VAL	OUT	Int	Codice della condizione di esecuzione

Funzionamento

L'istruzione SRT_DINT specifica un tempo di ritardo, avvia il timer interno per calcolarlo e associa il sottoprogramma di un OB di allarme all'evento di timeout del tempo di ritardo. Una volta trascorso il ritardo specificato, viene generato un allarme che attiva l'esecuzione dell'OB associato. Eseguendo l'istruzione CAN_DINT è possibile annullare un allarme di

ritardo inserito nel processo prima che scada il tempo di ritardo specificato. Il numero totale di eventi di ritardo e di schedulazione orologio attivi non deve essere superiore a quattro.

Inserimento nel progetto di sottoprogrammi di OB di allarme di ritardo

È possibile assegnare alle istruzioni SRT_DINT e CAN_DINT solo OB di allarme di ritardo. Poiché i nuovi progetti non ne contengono ancora l'utente dovrà inserirli nel seguente modo:

1. Fare doppio clic su "Inserisci nuovo blocco" nella diramazione "Blocchi di programma" dell'albero del progetto, selezionare "Blocco organizzativo (OB)" e scegliere "Allarmi di ritardo temporale".
2. A questo punto è possibile rinominare l'OB, selezionare il linguaggio di programmazione o scegliere il numero del blocco. Passare alla numerazione manuale se si desidera assegnare un numero di blocco diverso da quello assegnato automaticamente.
3. Modificare il sottoprogramma di OB di allarme di ritardo e programmare la reazione da eseguire quando si verifica l'evento di timeout del tempo di ritardo. Dall'OB di allarme di ritardo si possono richiamare altri blocchi di codice FC e FB fino a un massimo di quattro livelli di annidamento.
4. I nuovi nomi assegnati agli OB di allarme di ritardo risulteranno disponibili nel momento in cui si modifica il parametro OB_NR delle istruzioni SRT_DINT e CAN_DINT.

Codici delle condizioni di errore

RET_VAL (W#16#...)	Descrizione
0000	Nessun errore
8090	Parametro OB_NR errato
8091	Parametro DTIME errato
80A0	Allarme di ritardo non avviato

6.2.6.3 Istruzioni Disabilita allarmi e Abilita allarmi

Le istruzioni DIS_AIRT e EN_AIRT consentono di abilitare e disabilitare l'elaborazione degli allarmi.



DIS_AIRT ritarda l'elaborazione dei nuovi eventi di allarme. L'istruzione può essere eseguita più volte nello stesso OB e le esecuzioni vengono contate dal sistema operativo. Ogni esecuzione resta attiva finché non viene espressamente annullata da un'istruzione EN_AIRT o finché l'OB non è stato completamente elaborato.

Se riattivati, gli allarmi che si verificano durante l'esecuzione di DIS_AIRT vengono elaborati oppure vengono elaborati non appena si conclude l'esecuzione dell'OB attuale.



EN_AIRT abilita l'elaborazione degli eventi di allarme precedentemente disattivati con l'istruzione DIS_AIRT. Ogni esecuzione di DIS_AIRT deve essere annullata eseguendo un'istruzione EN_AIRT. Se, ad esempio, gli allarmi sono stati disattivati cinque volte con cinque esecuzioni di DIS_AIRT, per annullarle si dovrà eseguire cinque volte EN_AIRT.

Per poter riattivare gli allarmi per un dato OB è necessario che le esecuzioni di EN_AIRT vengano effettuate tutte all'interno di quell'OB o di un'FC o FB richiamati dallo stesso OB.

Il parametro RET_VAL indica quante volte è stata disattivata l'elaborazione degli allarmi e il suo valore corrisponde al numero di esecuzioni di DIS_AIRT messe in coda d'attesa.

L'elaborazione degli allarmi viene riattivata solo se il parametro RET_VAL = 0.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
RET_VAL	OUT	Int	Numero di ritardi = numero di esecuzioni di DIS_AIRT nella coda d'attesa.

6.2.7 Regolazione PID



L'istruzione "PID_Compact" realizza un regolatore PID con autotaratura per la modalità automatica e manuale.

Per maggiori informazioni sull'istruzione PID_Compact consultare la Guida in linea del portale TIA.

6.2.8 Istruzioni di controllo del movimento

Le istruzioni di controllo del movimento utilizzano un blocco dati tecnologico associato e le PTO dedicate (uscite di treni di impulsi) della CPU per comandare il movimento di un asse. Per maggiori informazioni sulle istruzioni di controllo del movimento consultare la Guida in linea di STEP 7 Basic.

ATTENZIONE

La frequenza massima degli impulsi dei generatori di impulsi è 100 KHz per le uscite digitali della CPU e 20 KHz per le uscite digitali della signal board. Tuttavia STEP 7 Basic non avvisa l'utente in caso di configurazione di un asse con una velocità o frequenza massima che supera questi limiti hardware. Ciò potrebbe causare problemi nell'applicazione, pertanto occorre sempre assicurarsi che non venga superata la frequenza massima degli impulsi consentita dall'hardware.



MC_Power attiva e disattiva un asse per il controllo del movimento.



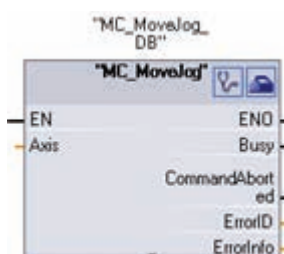
MC_Reset resetta tutti gli errori di controllo del movimento. Tutti gli errori di controllo del movimento confermabili vengono confermati.



MC_Home stabilisce la relazione tra il programma di comando e il sistema di posizionamento meccanico dell'asse.



MC_Halt annulla tutti i movimenti in corso arrestando il movimento dell'asse. La posizione di arresto non è definita.



MC_MoveJog attiva la modalità di marcia a impulsi a scopo di test e di avviamento.



MC_MoveAbsolute avvia il movimento verso una posizione assoluta. L'ordine termina quando viene raggiunta posizione di destinazione.



MC_MoveRelative avvia un movimento di posizionamento rispetto a una data posizione iniziale.



MC_MoveVelocity fa spostare l'asse alla velocità specificata.

Nota

Le uscite di treni di impulsi non possono essere utilizzate da altre istruzioni nel programma utente

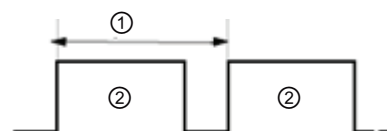
Quando si configurano le uscite della CPU o della signal board come generatori di impulsi (per l'utilizzo con le istruzioni PWM o di controllo base del movimento), gli indirizzi delle uscite corrispondenti (Q0.0, Q0.1, Q4.0 e Q4.1) vengono cancellati dalla memoria Q e non possono essere utilizzati per altri scopi nel programma utente. Se il programma utente scrive un valore in un'uscita utilizzata come generatore di impulsi, la CPU non scrive quel valore nell'uscita fisica.

6.2.9 Istruzione PWM

6.2.9.1 Istruzione CTRL_PWM

L'istruzione CTRL_PWM - Modulazione ampiezza impulsi (PWM) fornisce un'uscita con tempo di ciclo fisso e duty cycle variabile. Dopo essere stata avviata alla frequenza specificata (tempo di ciclo) l'uscita PWM continua a funzionare ininterrottamente.

La durata degli impulsi può essere variata in funzione del controllo desiderato.



① Tempo di ciclo

② Durata dell'impulso

La durata degli impulsi può essere espressa in centesimi del tempo di ciclo (0 – 100), in millesimi (0 – 1000), in decimillesimi (0 – 10000) oppure in formato analogico S7. La durata degli impulsi può variare da 0 (nessun impulso, sempre off) al valore di fondo scala (nessun impulso, sempre on).

Poiché può essere variata da 0 al valore di fondo scala, l'uscita PWM, pur essendo digitale, è molto simile a un'uscita analogica. La si può utilizzare, ad esempio, per comandare la velocità di un motore dalla posizione di arresto alla velocità massima o per comandare la posizione di una valvola da chiusa a completamente aperta.

Le uscite di impulsi veloci possono essere comandate con due generatori di impulsi: PWM e Pulse train output (PTO). Il PTO viene utilizzato dalle istruzioni di controllo del movimento. Ogni generatore di impulsi può essere assegnato a PWM o PTO ma non a entrambe contemporaneamente.

Come illustrato nella seguente tabella ai due generatori di impulsi sono assegnate specifiche uscite digitali. Si possono utilizzare le uscite onboard della CPU o quelle della signal board opzionale. La tabella riporta i numeri delle uscite che in questo caso corrispondono a quelli della configurazione di default. Se la numerazione delle uscite è stata modificata dall'utente i numeri saranno quelli da lui assegnati. Indipendentemente da ciò PTO1/PWM1 utilizza le prime due uscite digitali e PTO2/PWM2 le due successive, nella CPU o nella signal board. Si noti che PWM richiede una sola uscita, mentre PTO ne può utilizzare in opzione anche due per canale. Se un'uscita non è occupata da una funzione a impulsi può essere usata per altri scopi.

Descrizione	Assegnazione di default dell'uscita		
		Impulso	Direzione
PTO 1	CPU onboard	Q0.0	Q0.1
	Signal board	Q4.0	Q4.1
PWM 1	CPU onboard	Q0.0	--
	Signal board	Q4.0	--
PTO 2	CPU onboard	Q0.2	Q0.3
	Signal board	Q4.2	Q4.3
PWM 2	CPU onboard	Q0.2	--
	Signal board	Q4.2	--

Configurazione di un canale impulsivo per PWM

Per poter utilizzare l'operazione PWM è innanzitutto necessario configurare un canale impulsivo nella Configurazione dispositivi selezionando nell'ordine la CPU, il generatore di impulsi (PTO/PWM) e PWM1 o PWM2. Una volta attivato, il generatore di impulsi (casella di opzione) viene definito con un nome di default. Questo può essere modificato nella casella "Nome:" ma deve essere in ogni caso univoco. I nomi dei generatori di impulsi diventano variabili della tabella delle "Costanti" e potranno essere utilizzati come parametro PWM dell'istruzione CTRL_PWM.

ATTENZIONE

La frequenza massima degli impulsi dei generatori di impulsi è 100 KHz per le uscite digitali della CPU e 20 KHz per le uscite digitali della signal board. Tuttavia STEP 7 Basic non avvisa l'utente in caso di configurazione di un asse con una velocità o frequenza massima che supera questi limiti hardware. Ciò potrebbe causare problemi nell'applicazione, pertanto occorre sempre assicurarsi che non venga superata la frequenza massima degli impulsi consentita dall'hardware.

È possibile rinominare il generatore di impulsi, inserire un commento e assegnare i parametri come indicato nel paragrafo che segue.

- Generatore di impulsi come: PWM o PTO (scegliere PWM)
- Sorgente di uscita: CPU onboard o signal board
- Base di tempo: millisecondi o microsecondi
- Formato durata impulso:
 - Centesimi (da 0 a 100)
 - Millesimi (0 – 1000)
 - Decimillesimi (0 – 10000)
 - Formato analogico S7 (da 0 a 27648)
- Tempo di ciclo: indicare il valore del tempo di ciclo. Questo valore può essere modificato solo in Configurazione dispositivi.
- Durata impulso iniziale: indicare il valore della durata dell'impulso iniziale. Il valore può essere modificato durante il runtime.

Indirizzi di uscita



Indirizzo iniziale: indicare l'indirizzo di parola Q in cui si vuole inserire il valore della durata degli impulsi. L'indirizzo di default è QW1000 per PWM1 e QW1002 per PWM2. Il valore in questo indirizzo comanda la durata degli impulsi e viene inizializzato sul valore specificato in "Durata impulso iniziale:" ogni volta che il CPU commuta da STOP a RUN. Modificando questo valore di parola Q durante il runtime è possibile modificare la durata dell'impulso.

Parametro	Tipo di parametro	Tipo di dati	Valore iniziale	Descrizione
PWM	IN	WORD	0	Identificativo della PWM: i nomi dei generatori di impulsi attivati diventano variabili nella tabella "Costante" e possono essere utilizzati come parametro PWM.
ENABLE	IN	Bool		1 = avvia generatore di impulsi 0 = arresta generatore di impulsi
BUSY	OUT	Bool	0	Funzione occupata
STATUS	OUT	WORD	0	Codice della condizione di esecuzione

Funzionamento

L'istruzione CTRL_PWM utilizza un blocco dati (DB) per memorizzare le informazioni dei parametri. Quando si inserisce un'istruzione CTRL_PWM nell'editor di programma, viene assegnato automaticamente un DB i cui parametri non vengono modificati dall'utente ma comandati dall'istruzione stessa.

Specificare quale generatore di impulsi attivato si vuole utilizzare indicandone il nome di variabile come parametro PWM.

Se l'ingresso EN è vero l'istruzione PWM_CTRL avvia o arresta il PWM identificato in funzione del valore assunto dall'ingresso ENABLE. La durata degli impulsi è specificata dal valore contenuto nell'indirizzo di uscita di parola Q.

Poiché l'S7-1200 elabora la richiesta durante l'esecuzione di CTRL_PWM, nelle CPU S7-1200 il parametro BUSY è sempre "falso".

Se viene rilevato un errore, ENO viene impostato su "falso" e il codice della relativa condizione viene scritto nel parametro STATUS.

La durata degli impulsi viene impostata sul valore iniziale configurato in Configurazione dispositivi la prima volta che il PLC passa in RUN. I valori necessari per cambiare la durata degli impulsi vanno scritti nell'indirizzo di parola Q specificato in Configurazione dispositivi ("Indirizzi di uscita" / "Indirizzo iniziale:"). Per scrivere la durata degli impulsi nella parola Q appropriata si può usare un box di trasferimento, conversione, calcolo matematico o PID, rispettando il campo valido (percentuale, migliaia, decine di migliaia o formato analogico S7).

Codici delle condizioni di errore

Valore STATUS	Descrizione
0	Nessun errore
80A1	L'identificatore del PWM non indirizza un PWM valido

Impossibile forzare gli I/O digitali assegnati a PWM e PTO

Gli I/O digitali utilizzati dai dispositivi di modulazione dell'ampiezza degli impulsi (PWM) e di uscita di treni di impulsi (PTO) vengono assegnati durante la configurazione dei dispositivi. Quando sono assegnati a questi dispositivi indirizzi di I/O digitali, i valori di tali indirizzi non possono essere modificati dalla funzione di forzamento nella tabella di controllo.

Le uscite di treni di impulsi non possono essere utilizzate da altre istruzioni nel programma utente

Quando si configurano le uscite della CPU o della signal board come generatori di impulsi (per l'utilizzo con le istruzioni PWM o di controllo base del movimento), gli indirizzi delle uscite corrispondenti (Q0.0, Q0.1, Q4.0 e Q4.1) vengono cancellati dalla memoria Q e non possono essere utilizzati per altri scopi nel programma utente. Se il programma utente scrive un valore in un'uscita utilizzata come generatore di impulsi, la CPU non scrive quel valore nell'uscita fisica.

6.3 Istruzioni della biblioteca globale

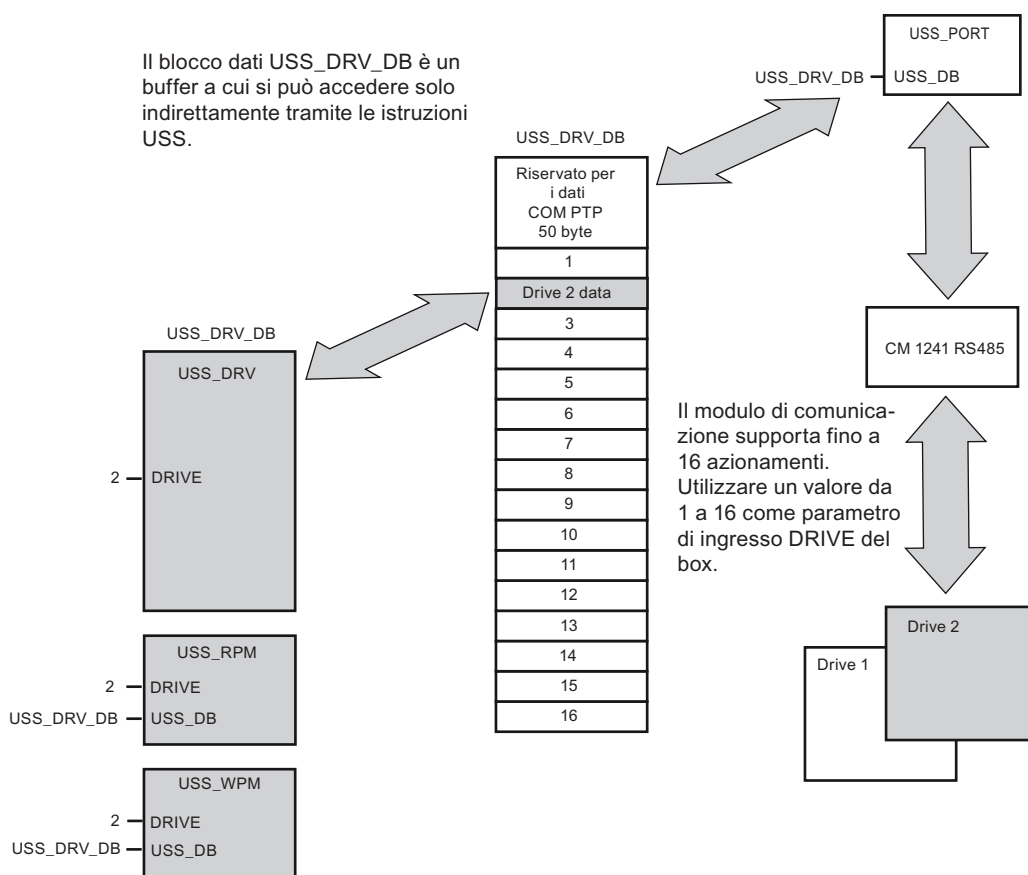
6.3.1 USS

La biblioteca del protocollo USS consente di comandare gli azionamenti Siemens che supportano il protocollo USS. Le istruzioni comprendono funzioni progettate specificatamente per comunicare con l'azionamento mediante il protocollo USS. Il modulo CM 1241 RS485 comunica con gli azionamenti attraverso le porte RS485. La biblioteca USS consente di comandare gli azionamenti fisici e di scriverne e leggerne i parametri.

6.3.1.1 Requisiti per l'utilizzo del protocollo USS

La biblioteca contiene 1 FB e 3 FC che supportano il protocollo USS. Ogni modulo di comunicazione CM 1241 RS485 supporta un massimo di 16 azionamenti.

Ogni singolo blocco dati di istanza contiene una memoria e un buffer temporanei per tutti gli azionamenti della rete USS collegati ai diversi moduli di comunicazione PtP installati. Le funzioni USS per questi azionamenti condividono le informazioni contenute nel blocco dati.



Gli azionamenti (fino a 16) collegati a un singolo CM 1241 RS485 fanno parte della stessa rete USS. Gli azionamenti collegati a un diverso CM 1241 RS485 fanno parte di una diversa rete USS. Poiché l'S7-1200 supporta fino a tre CM 1241 RS485, si possono avere fino a tre reti USS con 16 azionamenti ciascuna per un totale di 48 azionamenti USS.

Ogni rete USS viene gestita con un unico blocco dati (tre blocchi per tre reti USS che utilizzano tre CM 1241 RS485). Le istruzioni associate a una singola rete USS devono condividere il blocco dati. Tra queste figurano le istruzioni USS_DRV, USS_PORT, USS_RPM e USS_WPM utilizzate per controllare tutti gli azionamenti in una singola rete USS.

L'istruzione USS_DRV è un blocco funzionale (FB). Quando la si inserisce nell'editor la finestra di dialogo "Opzioni di richiamo" chiede quale DB assegnare all'FB. Se quella inserita è la prima istruzione USS_DRV inserita nel programma per la rete USS in questione, si può confermare il DB di default (o eventualmente rinominarlo) e verrà creato il nuovo DB. Se invece questa non è la prima istruzione USS_DRV per il canale in oggetto, si deve impostare il DB precedentemente assegnato alla rete USS selezionandolo nell'elenco di riepilogo della finestra "Opzioni di richiamo".

Le istruzioni USS_PORT, USS_RPM e USS_WPM sono tutte funzioni (FC). Quando si inseriscono nell'editor queste FC non viene assegnato alcun DB. Il DB adatto deve essere assegnato manualmente all'ingresso "USS_DB" delle istruzioni (fare doppio clic sul campo del parametro, quindi selezionare l'icona di aiuto per visualizzare i DB disponibili).

La funzione USS_PORT gestisce la comunicazione tra la CPU e gli azionamenti tramite il modulo di comunicazione PtP. Ogni suo richiamo gestisce la comunicazione con un azionamento. Il programma deve richiamarla abbastanza rapidamente da evitare che gli

azionamenti segnalino un timeout della comunicazione. La funzione può essere richiamata dall'OB principale o da qualsiasi OB di allarme.

Il blocco funzionale USS_DRV consente al programma di accedere a un azionamento specifico della rete USS. I suoi ingressi e le sue uscite corrispondono agli stati e alle funzioni di comando dell'azionamento. Se la rete contiene 16 azionamenti il programma deve avere almeno 16 richiami USS_DRV, uno per azionamento. Questi blocchi dovrebbero essere richiamati alla velocità necessaria per comandare le funzioni dell'azionamento.

Il blocco funzionale USS_DRV può essere richiamato solo dall'OB principale.

CAUTELA
--

USS_DRV, USS_RPM, USS_WPM devono essere richiamate solo dall'OB principale. La funzione USS_PORT può essere richiamata da qualsiasi OB, solitamente da un OB di allarme di ritardo.

Se non si riesce a evitare l'interruzione di USS_PORT si possono verificare errori imprevisti.
--

Le funzioni USS_RPM e USS_WPM leggono e scrivono i parametri di esercizio dell'azionamento remoto, i quali comandano il funzionamento interno dell'azionamento. Per maggiori informazioni vedere la descrizione dei parametri nel manuale dell'azionamento. Il programma può contenere tante funzioni quante ne sono necessarie ma, in un dato momento, può essere attiva una sola richiesta di lettura o di scrittura per azionamento. Le funzioni USS_RPM e USS_WPM possono essere richiamate solo da un OB principale.

Calcolo del tempo necessario per la comunicazione con l'azionamento

La comunicazione con l'azionamento è asincrona rispetto al ciclo di scansione dell'S7-1200. Generalmente l'S71200 esegue più cicli di scansione entro il tempo impiegato per concludere una transazione di comunicazione con l'azionamento.

L'intervallo USS_PORT è il tempo necessario per effettuare una transazione con l'azionamento. La tabella sotto illustrata indica l'intervallo USS_PORT minimo per ciascuna velocità di trasmissione. Se la funzione USS_PORT viene richiamata con una frequenza superiore a quella dell'intervallo USS_PORT, il numero di transazioni non aumenta. L'intervallo di timeout dell'azionamento è la quantità di tempo utilizzabile per una transazione nel caso in cui, a causa di errori di comunicazione, sia necessario effettuare tre tentativi per concluderla. Per default la biblioteca del protocollo USS effettua automaticamente fino a due tentativi per ogni transazione.

Velocità di trasmissione	Intervallo minimo calcolato tra i richiami di USS_PORT (millisecondi)	Intervallo di timeout per i messaggi degli azionamenti per azionamento (millisecondi)
1200	790	2370
2400	405	1215
4800	212.5	638
9600	116.3	349
19200	68.2	205
38400	44.1	133
57600	36.1	109
115200	28.1	85

6.3.1.2 Istruzione USS_DRV

L'istruzione USS_DRV scambia i dati con l'azionamento creando messaggi di richiesta e interpretando i messaggi di risposta dell'azionamento. Si deve utilizzare un blocco funzionale specifico per ogni azionamento, ma tutte le funzioni USS associate a una rete USS e a un modulo di comunicazione PtP devono impiegare lo stesso blocco dati di istanza. Quando si inserisce la prima istruzione USS_DRV si deve definire il nome del DB e in seguito riutilizzare sempre questo DB.

Durante la prima esecuzione di USS_DRV l'azionamento indicato dall'indirizzo USS (parametro DRIVE) viene inizializzato nel DB di istanza. Dopo l'inizializzazione le esecuzioni successive di USS_PORT possono avviare la comunicazione con l'azionamento che ha il numero specificato.

Se si modifica il numero dell'azionamento si deve commutare il PLC da STOP in RUN e inizializzare il DB di istanza. I parametri di ingresso vengono configurati nel buffer di trasmissione USS e le uscite vengono lette da un "precedente" buffer delle risposte valide (se presente). Mentre viene eseguita USS_DRV la trasmissione non viene effettuata. La comunicazione con gli azionamenti si attiva dopo l'esecuzione di USS_PORT. USS_DRV configura solamente i messaggi da trasmettere e interpreta i dati che potrebbero essere stati ricevuti da una precedente richiesta.

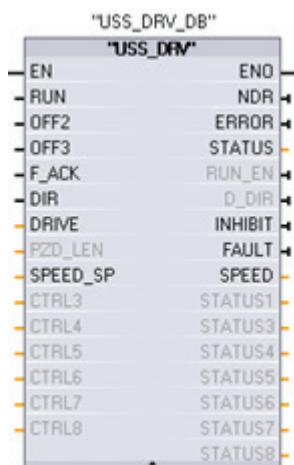
La direzione di rotazione dell'azionamento può essere controllata con l'ingresso DIR (BOOL) o utilizzando il segno (positivo o negativo) con l'ingresso SPEED_SP (REAL). La seguente tabella indica come gli ingressi interagiscono per determinare la direzione dell'azionamento, presupponendo che il motore sia cablato per la rotazione in avanti.

SPEED_SP	DIR	Direzione dell'azionamento
Valore > 0	0	Inversione
Valore > 0	1	Avanti
Valore < 0	0	Avanti
Valore < 0	1	Inversione

KOP (vista di default)



KOP (vista ampliata)



Facendo clic sulla freccia in basso si può espandere il box e visualizzarne tutti i parametri.

I parametri in grigio sono opzionali e non devono essere assegnati obbligatoriamente.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
RUN	IN	Bool	Bit di start dell'azionamento: se vero questo ingresso abilita l'azionamento a funzionare alla velocità preimpostata.
OFF2	IN	Bool	Bit di stop elettrico: se falso questo bit fa sì che l'azionamento si arresti per inerzia senza frenare.
OFF3	IN	Bool	Bit di stop rapido: se falso questo bit determina un arresto rapido frenando l'azionamento invece che arrestandolo per inerzia.
F_ACK	IN	Bool	Bit di riconoscimento errori: questo bit resetta il bit di errore in un azionamento e viene impostato dopo che l'errore è stato eliminato, in modo che l'azionamento sappia che non deve più segnalare l'errore.
DIR	IN	Bool	Bit di controllo dell'azionamento: questo bit viene impostato per indicare che la direzione è "in avanti" (se SPEED_SP è positivo).
DRIVE	IN	USInt	Indirizzo dell'azionamento: questo ingresso è l'indirizzo dell'azionamento USS. Il campo valido va dall'azionamento 1 all'azionamento 16.
PZD_LEN	IN	USInt	Lunghezza in parole: è il numero di parole richiesto dai dati PZD. Sono validi i valori 2, 4, 6 o 8 (parole). Per default è impostato 2.
SPEED_SP	IN	Real	Setpoint della velocità: è la velocità dell'azionamento espressa come percentuale della frequenza configurata. Un valore positivo indica la direzione in avanti (se DIR è vero).
CTRL3	IN	UInt	Parola di comando 3: valore scritto in un parametro dell'azionamento configurabile dall'utente. La configurazione deve essere effettuata nell'azionamento. Il parametro è opzionale.

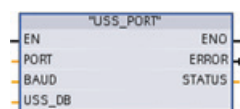
Parametro	Tipo di parametro	Tipo di dati	Descrizione
CTRL4	IN	UInt	Parola di comando 4: valore scritto in un parametro dell'azionamento configurabile dall'utente. La configurazione deve essere effettuata nell'azionamento. Il parametro è opzionale.
CTRL5	IN	UInt	Parola di comando 5: valore scritto in un parametro dell'azionamento configurabile dall'utente. La configurazione deve essere effettuata nell'azionamento. Il parametro è opzionale.
CTRL6	IN	UInt	Parola di comando 6: valore scritto in un parametro dell'azionamento configurabile dall'utente. La configurazione deve essere effettuata nell'azionamento.
CTRL7	IN	UInt	Parola di comando 7: valore scritto in un parametro dell'azionamento configurabile dall'utente. La configurazione deve essere effettuata nell'azionamento. Il parametro è opzionale.
CTRL8	IN	UInt	Parola di comando 8: valore scritto in un parametro dell'azionamento configurabile dall'utente. La configurazione deve essere effettuata nell'azionamento. Il parametro è opzionale.
NDR	OUT	Bool	Nuovi dati disponibili: se vero il bit indica che le uscite contengono i dati di una nuova richiesta di comunicazione.
ERROR	OUT	Bool	Si è verificato un errore: se vero indica che si è verificato un errore e che lo stato (STATUS) dell'uscita è valido. In caso di errore le altre uscite vengono impostate a 0. Gli errori di comunicazione vengono segnalati solo nelle uscite ERROR e STATUS dell'istruzione USS_PORT.
STATUS	OUT	UInt	Valore di stato della richiesta. Indica il risultato della scansione Non è una parola di stato restituita dall'azionamento.
RUN_EN	OUT	Bool	Esecuzione abilitata: questo bit indica se l'azionamento è in funzione.
D_DIR	OUT	Bool	Direzione dell'azionamento: questo bit indica se la direzione di funzionamento dell'azionamento è "in avanti".
INHIBIT	OUT	Bool	Azionamento inibito: questo bit indica lo stato del bit di inibizione nell'azionamento.
FAULT	OUT	Bool	Errore dell'azionamento: questo bit indica che l'azionamento ha registrato un errore. Per resettare il bit l'utente deve risolvere il problema e impostare il bit F_ACK.
SPEED	OUT	REAL	Velocità attuale dell'azionamento (valore in scala della parola di stato dell'azionamento 2): valore della velocità dell'azionamento espresso come percentuale della velocità configurata.
STATUS1	OUT	UInt	Parola di stato dell'azionamento 1: questo valore contiene i bit di stato fissi di un azionamento.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
STATUS3	OUT	UInt	Parola di stato dell'azionamento 3: questo valore contiene una parola di stato configurabile dall'utente nell'azionamento.
STATUS4	OUT	UInt	Parola di stato dell'azionamento 4: questo valore contiene una parola di stato configurabile dall'utente nell'azionamento.
STATUS5	OUT	UInt	Parola di stato dell'azionamento 5: questo valore contiene una parola di stato configurabile dall'utente nell'azionamento.
STATUS6	OUT	UInt	Parola di stato dell'azionamento 6: questo valore contiene una parola di stato configurabile dall'utente nell'azionamento.
STATUS7	OUT	UInt	Parola di stato dell'azionamento 7: questo valore contiene una parola di stato configurabile dall'utente nell'azionamento.
STATUS8	OUT	UInt	Parola di stato dell'azionamento 8: questo valore contiene una parola di stato configurabile dall'utente nell'azionamento.

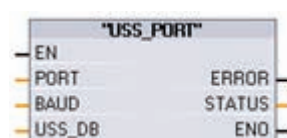
6.3.1.3 Istruzione USS_PORT

L'istruzione USS_PORT gestisce la comunicazione attraverso la rete USS. Generalmente il programma contiene una sola funzione USS_PORT per modulo di comunicazione PtP e ogni richiamo della funzione gestisce una trasmissione da o verso un solo azionamento. Il programma deve eseguire la funzione USS_PORT un numero di volte sufficiente a evitare che si verifichino timeout nell'azionamento. Tutte le funzioni USS associate a una rete USS e a un modulo di comunicazione PtP devono utilizzare lo stesso blocco dati di istanza. USS_PORT viene generalmente richiamata da un OB di allarme di ritardo al fine di evitare timeout dell'azionamento e mantenere i dati USS più recenti a disposizione dei richiami dell'istruzione USS_DRV.

KOP



FUP

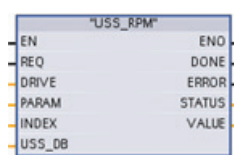


Parametro	Tipo di parametro	Tipo di dati	Descrizione
PORT	IN	Porta	Modulo di comunicazione PtP. Identificativo: è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.
BAUD	IN	DInt	Velocità della comunicazione USS.
USS_DB	IN	DInt	Riferimento al DB di istanza creato e inizializzato in seguito all'inserimento di un'istruzione USS_DRV nel programma.

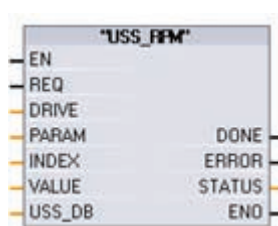
Parametro	Tipo di parametro	Tipo di dati	Descrizione
ERROR	OUT	Bool	Se vero questo pin indica che si è verificato un errore e che lo stato (STATUS) dell'uscita è valido.
STATUS	OUT	UInt	Valore di stato della richiesta. Indica il risultato della scansione o inizializzazione. Maggiori informazioni sono disponibili nella variabile "USS_Extended_Error" per alcuni codici di stato.

6.3.1.4 Istruzione USS_RPM

KOP



FUP



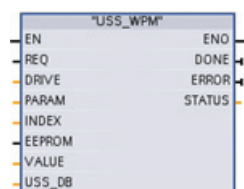
L'istruzione USS_RPM legge un parametro dall'azionamento. Tutte le funzioni USS associate a una rete USS e a un modulo di comunicazione PtP devono utilizzare lo stesso blocco dati. USS_RPM deve essere richiamata dall'OB principale.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Richiesta di invio: se vero indica che è presente una nuova richiesta di lettura. Viene ignorato se è già presente una richiesta.
DRIVE	IN	USInt	Indirizzo dell'azionamento: questo ingresso è l'indirizzo dell'azionamento USS. Il campo valido va dall'azionamento 1 all'azionamento 16.
PARAM	IN	UInt	Numero del parametro: questo ingresso indica quale parametro dell'azionamento viene scritto. Il campo per il parametro va da 0 a 2047. Per informazioni su come accedere ai parametri con valori maggiori di questo campo vedere il manuale dell'azionamento.
INDEX	IN	UInt	Indice del parametro: questo ingresso indica quale indice del parametro dell'azionamento deve essere scritto. È un valore di 16 bit il cui byte meno significativo costituisce il valore effettivo dell'indice (compreso entro un campo da 0 a 255). Il byte più significativo può essere usato anche dall'azionamento ed è specifico di quest'ultimo. Per maggiori informazioni consultare il manuale dell'azionamento.
USS_DB	IN	Versione	Riferimento al DB di istanza creato e inizializzato in seguito all'inserimento di un'istruzione USS_DRV nel programma.
VALUE	IN	Word, Int, UInt, DWord, DInt, UInt, Real	Valore del parametro letto, è valido solo se il bit DONE è vero.

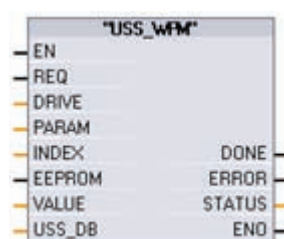
Parametro	Tipo di parametro	Tipo di dati	Descrizione
DONE	OUT	Bool	Concluso: se vero indica che l'uscita VALUE contiene il valore del parametro di lettura richiesto precedentemente. Il bit viene impostato quando USS_DRV rileva i dati della risposta di lettura dall'azionamento. Il bit viene resettato: - se i dati della risposta sono stati richiesti con un'altra interrogazione USS_RPM - oppure - durante il secondo di due richiami successivi dell'istruzione USS_DRV
ERROR	OUT	Bool	Si è verificato un errore: se vero indica che si è verificato un errore e che lo stato (STATUS) dell'uscita è valido. In caso di errore le altre uscite vengono impostate a 0. Gli errori di comunicazione vengono segnalati solo nelle uscite ERROR e STATUS dell'istruzione USS_PORT.
STATUS	OUT	UInt	Valore di stato della richiesta. Indica il risultato della richiesta di lettura. Maggiori informazioni sono disponibili nella variabile "USS_Extended_Error" per alcuni codici di stato.

6.3.1.5 Istruzione USS_WPM

KOP



FUP



L'istruzione USS_WPM modifica un parametro dell'azionamento. Tutte le funzioni USS associate a una rete USS e a un modulo di comunicazione PtP devono utilizzare lo stesso blocco dati. USS_WPM deve essere richiamata dall'OB principale.

Nota

Operazioni di scrittura nella EEPROM

È importante non utilizzare eccessivamente l'operazione di scrittura permanente nella EEPROM e limitare il più possibile l'utilizzo della memoria per aumentarne la durata.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Richiesta di invio: se vero indica che è presente una nuova richiesta di lettura. Viene ignorato se è già presente una richiesta.
DRIVE	IN	USInt	Indirizzo dell'azionamento: questo ingresso è l'indirizzo dell'azionamento USS. Il campo valido va dall'azionamento 1 all'azionamento 16.
PARAM	IN	UInt	Numero del parametro: questo ingresso indica quale parametro dell'azionamento viene scritto. Il campo per il parametro va da 0 a 2047. Per informazioni su come accedere ai parametri con valori maggiori di questo campo vedere il manuale dell'azionamento.
INDEX	IN	UInt	Indice del parametro: questo ingresso indica quale indice del parametro dell'azionamento deve essere scritto. È un valore di 16 bit il cui byte meno significativo costituisce il valore effettivo dell'indice (compreso entro un campo da 0 a 255). Il byte più significativo può essere usato anche dall'azionamento ed è specifico di quest'ultimo. Per maggiori informazioni consultare il manuale dell'azionamento.
EEPROM	IN	Bool	Salva nella EEPROM dell'azionamento: se vero le scritture nel parametro dell'azionamento vengono salvate nella EEPROM dell'azionamento. Se falso la scrittura è temporanea e non viene mantenuta in caso di spegnimento e riaccensione dell'azionamento.
VALUE	IN	Word, Int, UInt, DWord, DInt, UDIInt, Real	Valore del parametro da scrivere. Deve essere valido durante la commutazione dello stato di REQ.
USS_DB	IN	Versione	Riferimento al DB di istanza creato e inizializzato in seguito all'inserimento di un'istruzione USS_DRV nel programma.
DONE	OUT	Bool	Concluso: se vero indica che l'ingresso VALUE è stato scritto nell'azionamento. Il bit viene impostato quando USS_DRV rileva i dati della risposta di scrittura dall'azionamento. Il bit viene resettato: per richiedere all'azionamento la conferma che la scrittura è stata effettuata l'utente utilizza un'altra interrogazione USS_WPM oppure il secondo di due richiami successivi dell'istruzione.
ERROR	OUT	Bool	Si è verificato un errore: se vero indica che si è verificato un errore e che lo stato (STATUS) dell'uscita è valido. In caso di errore le altre uscite vengono impostate a 0. Gli errori di comunicazione vengono segnalati solo nelle uscite ERROR e STATUS dell'istruzione USS_PORT.
STATUS	OUT	UInt	Valore di stato della richiesta. Indica il risultato della richiesta di scrittura. Maggiori informazioni sono disponibili nella variabile "USS_Extended_Error" per alcuni codici di stato.

6.3.1.6 Codici di stato USS

I codici di stato dell'istruzione USS vengono restituiti nell'uscita STATUS delle funzioni USS.

Valore STATUS (W#16#....)	Descrizione
0000	Nessun errore
8180	La lunghezza della risposta dell'azionamento non corrisponde ai caratteri ricevuti dall'azionamento. Il numero dell'azionamento in cui si è verificato l'errore viene restituito nella variabile "USS_Extended_Error". Vedere la tabella degli errori avanzati riportata più avanti.
8181	Il parametro VALUE non era di tipo Word, Real o DWord.
8182	L'utente ha specificato un parametro in formato Word e ha ricevuto dall'azionamento una risposta in formato DWord o Real
8183	L'utente ha specificato un parametro in formato DWord o Real e ha ricevuto dall'azionamento una risposta in formato Word
8184	La somma di controllo nel telegramma di risposta dall'azionamento era errata. Il numero dell'azionamento in cui si è verificato l'errore viene restituito nella variabile "USS_Extended_Error". Vedere la tabella degli errori avanzati riportata più avanti.
8185	Indirizzo dell'azionamento non ammesso (campo valido: 1-16)
8186	Setpoint della velocità non compreso entro il campo valido (da -200% a 200%)
8187	La risposta alla richiesta trasmessa è arrivata dal numero di azionamento errato. Il numero dell'azionamento in cui si è verificato l'errore viene restituito nella variabile "USS_Extended_Error". Vedere la tabella degli errori avanzati riportata più avanti.
8188	La lunghezza specificata in PZD non è ammessa (campo valido = 2, 4, 6 o 8 parole)
8189	È stata specificata una velocità di trasmissione non ammessa
818A	Il canale per la richiesta dei parametri è utilizzato da un'altra richiesta diretta a questo azionamento
818B	L'azionamento non ha risposto alle richieste e ai tentativi di ritrasmissione. Il numero dell'azionamento in cui si è verificato l'errore viene restituito nella variabile "USS_Extended_Error". Vedere la tabella degli errori avanzati riportata più avanti.
818C	L'azionamento ha restituito un errore avanzato a un'operazione di richiesta di parametri. Vedere la tabella degli errori avanzati riportata più avanti.
818D	L'azionamento ha restituito un errore di accesso non ammesso per un'operazione di richiesta di parametri. Per maggiori informazioni sul motivo per cui l'accesso ai parametri potrebbe essere limitato consultare il manuale dell'azionamento
818E	L'azionamento non è stato inizializzato: questo codice di errore viene restituito alle istruzioni USS_RPM o USS_WPM se l'istruzione USS_DRV non è stata richiamata almeno una volta per l'azionamento in oggetto. In questo modo si evita che l'inizializzazione nel primo ciclo di USS_DRV sovrascriva una richiesta in attesa di scrittura o di lettura di parametri, poiché l'azionamento viene inizializzato come nuovo elemento. Per eliminare questo errore richiamare USS_DRV per il numero dell'azionamento in oggetto.
80Ax-80Fx	Errori specifici restituiti dagli FB di comunicazione PtP (punto a punto) richiamati dalla biblioteca USS: i valori di questi codici di errore non vengono modificati dalla biblioteca USS e sono definiti nelle descrizioni delle istruzioni PtP.

Codici degli errori avanzati degli azionamenti USS

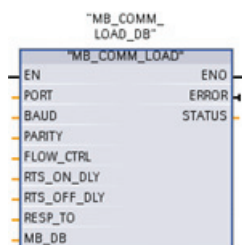
Gli azionamenti USS supportano l'accesso in lettura e in scrittura ai parametri interni dell'azionamento. Questa funzione consente di comandare e configurare l'azionamento a distanza. Le operazioni di accesso ai parametri dell'azionamento possono non riuscire a causa di errori quali "valori non compresi entro il campo" o "richieste non ammesse per la

modalità attuale dell'azionamento". L'azionamento genera il valore di un codice di errore che viene restituito nella variabile "USS_Extended_Error" del DB di istanza dell'istruzione USS_DRV. Il valore è valido solo per l'ultima esecuzione dell'istruzione USS_RPM o USS_WPM. Il codice di errore dell'azionamento viene inserito nella variabile "USS_Extended_Error" quando STATUS assume il valore esadecimale 818C. Il valore del codice di errore indicato in "USS_Extended_Error" dipende dal modello dell'azionamento. Per informazioni sui codici degli errori avanzati per le operazioni di lettura e scrittura dei parametri consultare il manuale dell'azionamento.

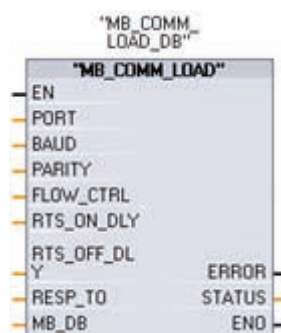
6.3.2 MODBUS

6.3.2.1 MB_COMM_LOAD

KOP



FUP



L'istruzione MB_COMM_LOAD configura una porta del modulo punto a punto (PtP) CM 1241 RS485 o CM 1241 RS232 per la comunicazione tramite protocollo Modbus RTU.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
PORT	IN	UInt	Identificativo della porta di comunicazione: dopo aver installato il modulo CM in Configurazione dispositivi l'identificativo della porta compare nell'elenco a discesa del collegamento PORT del box. Questa costante può essere anche indirizzata nella scheda "Costanti" della tabella delle variabili di default.
BAUD	IN	UDInt	Selezione della velocità di trasmissione: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200 Qualsiasi altro valore non è valido.
PARITY	IN	UInt	Parità: 0 – Nessuna parità 1 – Parità dispari 2 – Parità pari
FLOW_CTRL	IN	UInt	Controllo flusso: 0 – (default) Nessun controllo del flusso 1 – Controllo del flusso hardware con RTS sempre ON (non vale per le porte RS485) 2 - Controllo del flusso hardware con RTS sempre disattivato

Parametro	Tipo di parametro	Tipo di dati	Descrizione
RTS_ON_DLY	IN	UInt	Ritardo RTS ON: 0 – (default) Nessun ritardo da "RTS ON" alla trasmissione del primo carattere del messaggio - Da 1 a 65535 – Ritardo in millisecondi da "RTS ON" alla trasmissione del primo carattere del messaggio (non vale per le porte RS-485). I ritardi RTS vengono applicati indipendentemente dal tipo di controllo del flusso selezionato in FLOW_CTRL.
RTS_OFF_DLY	IN	UInt	Ritardo RTS OFF: 0 – (default) Nessun ritardo fra la trasmissione dell'ultimo carattere e la disattivazione di RTS - Da 1 a 65535 – Ritardo in millisecondi fra la trasmissione dell'ultimo carattere e la disattivazione di RTS (non vale per le porte RS-485). I ritardi RTS vengono applicati indipendentemente dal tipo di controllo del flusso selezionato in FLOW_CTRL.
RESP_TO	IN	UInt	Timeout di risposta: tempo in millisecondi concesso dal master MB_MASTER per la risposta dello slave. Se lo slave non risponde entro il tempo indicato il master MB_MASTER ripete la richiesta oppure, nel caso sia stato raggiunto il numero di tentativi specificati, la interrompe con un errore. Da 5 ms a 65535 ms (valore di default = 1000 ms).
MB_DB	IN	Versione	Riferimento al blocco dati di istanza utilizzato dalle istruzioni MB_MASTER o MB_SLAVE. Dopo aver inserito nel programma MB_SLAVE o MB_MASTER, l'identificativo del DB compare nell'elenco a discesa del collegamento MB_DB del box.
ERROR	OUT	Bool	Errore: 0 – Nessun errore rilevato 1 – Indica che è stato rilevato un errore e che il relativo codice indicato nel parametro STATUS è valido.
STATUS	OUT	WORD	Codice dell'errore di configurazione della porta

MB_COMM_LOAD consente di configurare una porta per il protocollo Modbus RTU. Una volta configurata la porta è possibile comunicare tramite Modbus con le istruzioni MB_SLAVE o MB_MASTER.

MB_COMM_LOAD deve essere richiamata una volta per inizializzare la porta. È necessario richiamarla solo se uno dei parametri di comunicazione deve essere modificato. MB_COMM_LOAD può essere richiamata da un OB di avvio ed eseguita una volta oppure si può avviare un richiamo che la esegua una volta utilizzando il flag di sistema del primo ciclo di scansione.

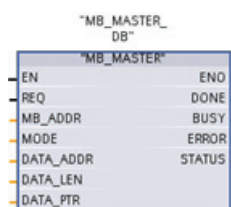
È necessario usare un'istanza diversa di MB_COMM_LOAD per ogni porta di ogni modulo riservato alla comunicazione Modbus e assegnare un DB di istanza MB_COMM_LOAD univoco per ogni porta utilizzata. La CPU S7-1200 può essere collegata al massimo a 3 moduli di comunicazione.

Quando si inserisce un'istruzione MB_MASTER o MB_SLAVE viene assegnato un blocco dati di istanza. È questo DB di istanza che deve essere indicato quando si specifica il parametro MB_DB nell'istruzione MB_COMM_LOAD.

Valore STATUS (W#16#....)	Descrizione
0000	Nessun errore
8180	Valore di ID della porta non valido
8181	Valore della velocità di trasmissione non valido
8182	Valore di parità non valido
8183	Valore di controllo del flusso non valido
8184	Valore del timeout di risposta non valido
8185	Puntatore MB_DB al DB di istanza per un MB_MASTER o MB_SLAVE non valido

6.3.2.2 MB_MASTER

KOP



FUP



L'istruzione MB_MASTER consente al programma di comunicare come master Modbus utilizzando una porta del modulo punto a punto (PtP) CM 1241 RS485 o CM 1241 RS232. È possibile accedere ai dati di uno o alcuni slave Modbus.

Quando si inserisce nel programma un'istruzione MB_MASTER viene assegnato un blocco dati di istanza. Il nome del DB di istanza di MB_MASTER va indicato quando si specifica il parametro MB_DB nell'istruzione MB_COMM_LOAD.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Ingresso richiesta: 0 – Nessuna richiesta 1 – Richiesta di trasmissione dei dati allo/agli slave Modbus
MB_ADR	IN	USInt	Indirizzo della stazione Modbus RTU: Campo di indirizzi valido: da 0 a 247 Il valore 0 è riservato alla trasmissione di un messaggio a tutti gli slave Modbus. Gli unici codici delle funzioni Modbus supportati per la trasmissione sono 05, 06, 15 e 16.
MODE	IN	USInt	Selezione del modo: Specifica il tipo di richiesta: lettura, scrittura o diagnostica Per maggiori informazioni vedere la tabella delle funzioni Modbus.
DATA_ADDR	IN	UDInt	Indirizzo iniziale nello slave: specifica l'indirizzo iniziale dei dati a cui si vuole accedere nello slave Modbus. Gli indirizzi validi sono elencati nella tabella delle funzioni Modbus riportata più avanti.
DATA_LEN	IN	UInt	Lunghezza dei dati: specifica il numero di bit o di parole a cui si deve accedere nella richiesta. Le lunghezze valide sono indicate nella tabella delle funzioni Modbus riportata più avanti.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
DATA_PTR	IN	Versione	Puntatore dati: punta all'indirizzo del DB della CPU per rilevare i dati da scrivere o leggere. Il DB deve essere di tipo "NON indirizzabile soltanto simbolicamente". Vedere l'avvertenza su DATA_PTR più avanti.
NDR	OUT	Bool	Nuovi dati disponibili: • 0 – Transazione non conclusa • 1 – Indica che l'istruzione MB_MASTER ha concluso la transazione richiesta con lo/gli slave Modbus
BUSY	OUT	Bool	Occupato. • 0 – Non è in corso alcuna transazione MB_MASTER • 1 – È in corso una transazione MB_MASTER
ERROR	OUT	Bool	Errore: • 0 – Nessun errore rilevato • 1 – Indica che è stato rilevato un errore e che il relativo codice indicato nel parametro STATUS è valido
STATUS	OUT	WORD	Codice della condizione di esecuzione

Regole per la comunicazione master Modbus

- Perché un'istruzione MB_MASTER possa comunicare con una porta è necessario che questa sia stata configurata eseguendo MB_COMM_LOAD.
- Se una porta viene utilizzata per inizializzare le richieste del master Modbus, non può essere utilizzata da MB_SLAVE. La porta può essere usata con una o più istanze di MB_MASTER.
- Le istruzioni Modbus non utilizzano eventi di allarme per comandare il processo di comunicazione. Per rilevare le condizioni di trasmissione e ricezione conclusa il programma deve interrogare l'istruzione MB_MASTER.
- Se il programma gestisce un master Modbus e utilizza MB_MASTER per trasmettere una richiesta a uno slave, si deve continuare a interrogare lo slave (eseguire MB_MASTER) finché non restituisce la risposta.
- L'esecuzione di MB_MASTER per una data porta deve essere sempre richiamata dallo stesso OB (o livello di priorità dell'OB).

Parametro REQ

Valore di REQ "falso" = nessuna richiesta

Valore di REQ "vero" = richiesta di trasmissione dei dati allo/agli slave Modbus

Questo valore deve essere fornito da un contatto attivato da un fronte positivo al primo richiamo dell'istruzione MB_MASTER. L'impulso attivato dal fronte richiederà una volta la richiesta di trasmissione. Tutti gli ingressi vengono rilevati e mantenuti invariati durante la richiesta e la risposta attivata da questo ingresso.

Internamente MB_MASTER avvierà una macchina a stati in modo da garantire che nessun altro MB_MASTER possa inviare una richiesta finché non è conclusa quella in corso.

Inoltre, se la stessa istanza del richiamo dell'FB MB_MASTER viene eseguita nuovamente con l'ingresso REQ = vero prima che la richiesta sia stata elaborata, non vengono effettuate

altre trasmissioni. Non appena la richiesta in corso si conclude ne viene trasmessa una nuova se MB_MASTER viene eseguita con l'ingresso REQ = vero.

I parametri DATA_ADDR e MODE consentono di selezionare il tipo di funzione Modbus

DATA_ADDR (indirizzo Modbus iniziale nello slave): specifica l'indirizzo iniziale dei dati a cui si vuole accedere nello slave Modbus.

MB_MASTER utilizza l'ingresso MODE invece che l'ingresso di un codice funzione. La combinazione di MODE e del campo degli indirizzi Modbus determina quale codice funzione verrà utilizzato nel messaggio Modbus. La seguente tabella descrive l'interazione tra il parametro MODE, il codice funzione e il campo degli indirizzi Modbus dell'istruzione MBUS_MASTER.

Funzioni di Modbus MB_MASTER				
	Parametro dell'indirizzo Modbus DATA_ADDR	Tipo di indirizzo	Parametro della lunghezza dei dati Modbus DATA_LEN	Funzione Modbus
Modo 0				
Lettura	00001 ... 09999	Bit di uscita	1 ... 2000	01H
	10001 – 19999	Bit di ingresso	1 ... 2000	02H
	30001 - 39999	Registri di ingresso	1 ... 125	04H
	40001 ... 49999	Registri di mantenimento	1 ... 125	03H
	400001 ... 465536 (avanzato)			
Modo 1				
Scrittura	00001 ... 09999	Bit di uscita	1 (bit singolo)	05H
	40001 ... 49999	Registri di mantenimento	1 (parola singola)	06H
	400001 ... 465536 (avanzato)			
	00001 ... 09999	Bit di uscita	2 ... 1968	15H
	40001 ... 49999	Registri di mantenimento	2 ... 123	16H
	400001 ... 465536 (avanzato)			
Modo 2				
Alcuni slave Modbus non supportano le scritture di singoli bit o parole con le funzioni Modbus 05H e 06H. In questi casi viene utilizzato il Modo 2 per forzare scritture di singoli bit o parole mediante le funzioni Modbus 15H e 16H.				
Scrittura	00001 ... 09999	Bit di uscita	1 ... 1968	15H
	40001 ... 49999	Registri di mantenimento	1 ... 123	16H
	400001 ... 465536 (avanzato)			
Modo 11				
• Legge una parola del contatore degli eventi dallo slave Modbus specificato come ingresso a MB_ADDR • In uno slave Modbus Siemens S7-1200, questo contatore viene incrementato ogni volta che lo slave riceve una richiesta di lettura o di scrittura valida (non una richiesta di trasmissione) da un master Modbus. • Il valore restituito viene memorizzato nell'indirizzo di parola specificato come ingresso a DATA_PTR. • Questo modo non richiede un valore DATA_LEN valido.				

Funzioni di Modbus MB_MASTER
Modo 80 • Controlla lo stato di comunicazione dello slave Modbus indicato come ingresso a MB_ADDR • L'impostazione del bit di uscita NDR nell'istruzione MB_MASTER indica che lo slave Modbus indirizzato ha risposto con i dati appropriati. • Non sono stati restituiti dati al programma. • Questo modo non richiede un valore DATA_LEN valido.
Modo 81 • Resetta il contatore degli eventi (restituito dal Modo 11) nello slave Modbus indirizzato come ingresso a MB_ADDR • L'impostazione del bit di uscita NDR nell'istruzione MB_MASTER indica che lo slave Modbus indirizzato ha risposto con i dati appropriati. • Non sono stati restituiti dati al programma. • Questo modo non richiede un valore DATA_LEN valido.

Parametro DATA_PTR

Il parametro DATA_PTR punta all'indirizzo di origine o di destinazione locale (nella CPU S7-1200) rispettivamente dei dati scritti o letti. Se si utilizza l'istruzione MB_MASTER per creare un master Modbus è necessario creare un blocco dati globale che memorizzi i dati per le letture e le scritture negli slave Modbus.

Nota

Il parametro DATA_PTR deve far riferimento a un blocco dati globale per il quale è stato attivato l'attributo Indirizzabile soltanto simbolicamente.

Quando si aggiunge un nuovo blocco dati per creare un DB di questo tipo è necessario deselezionare la casella "Indirizzabile soltanto simbolicamente".

Strutture di blocchi dati per il parametro DATA_PTR

- Questi tipi di dati sono validi per le **letture di parole** degli indirizzi Modbus da 30001 a 39999, da 40001 a 49999 e da 400001 a 465536 oltre che per le **scritture di parole** negli indirizzi Modbus da 40001 a 49999 e da 400001 a 465536.
 - Array standard con tipo di dati WORD, UINT o INT come indicato più avanti.
 - Struttura WORD, UINT o INT definita da un nome nella quale ciascun elemento ha un nome univoco e un tipo di dati di 16 bit.
 - Struttura complessa definita da un nome nella quale ciascun elemento ha un nome univoco e un tipo di dati di 16 o 32 bit.
- Per la lettura e la scrittura dei bit degli indirizzi Modbus da 00001 a 09999 e da 10001 a 19999.
 - Array standard con tipi di dati booleani.
 - Struttura booleana definita da un nome costituita da variabili booleane denominate in modo univoco.
- Nonostante non sia necessario è consigliabile associare ogni istruzione MB_MASTER a un'area specifica di un blocco dati globale. Questo perché se più istruzioni MB_MASTER

leggono e scrivono nella stessa area di un DB globale è più probabile che i dati si corrompano.

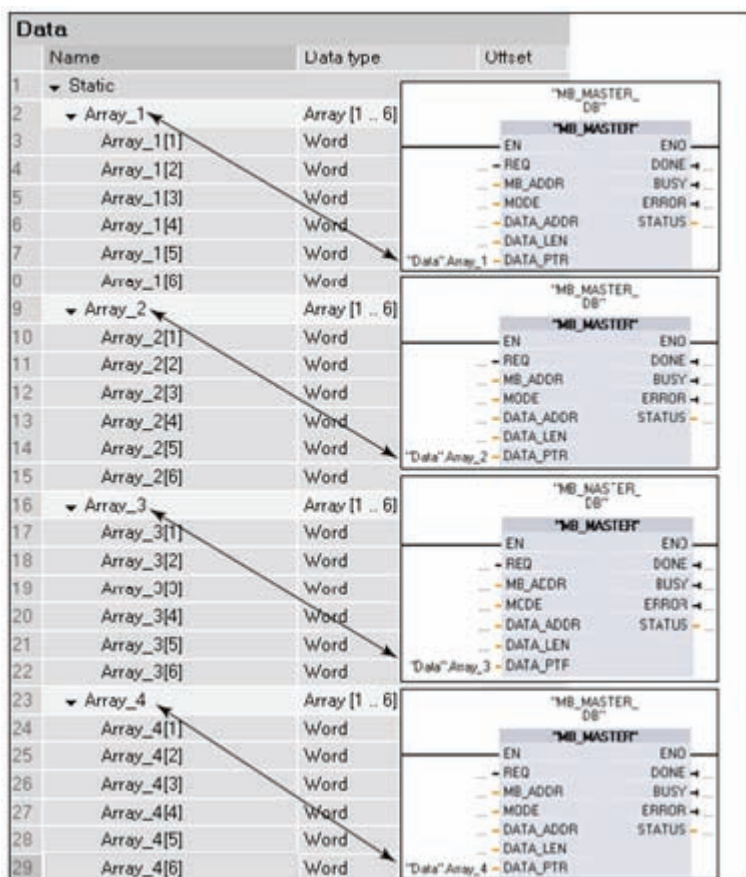
- Non è necessario che le aree di dati di DATA_PTR si trovino nello stesso blocco dati globale. È possibile creare un blocco dati con più aree per le letture Modbus, un blocco dati per le scritture Modbus o un blocco dati per ogni stazione slave.
- Tutti gli array dell'esempio sotto riportato sono stati creati come array a base 1 [1 ... ##]. Gli array potrebbero essere stati creati come array a base 0 [0 ... ###] o a base mista 0 e 1.

Esempio di istruzioni MB_MASTER che accedono a blocchi dati globali DATA_PTR

L'esempio di blocco dati globale sotto indicato utilizza 4 array di 6 parole con nomi univoci per la memorizzazione dei dati delle richieste Modbus. Gli array di dati possono avere qualsiasi dimensione, nonostante quelli dell'esempio siano uguali. Ogni array potrebbe anche essere sostituito con una struttura di dati costituita da nomi di variabili più descrittivi e tipi di dati misti. Esempi di strutture di dati alternative sono riportati nella descrizione del parametro HR_DB dell'istruzione MB_SLAVE (Pagina 212).

Gli esempi di istruzione MB_MASTER sotto illustrati mostrano solo il parametro DATA_PTR e non gli altri parametri richiesti e hanno lo scopo di illustrare come l'istruzione MB_MASTER utilizza il blocco dati DATA_PTR.

Le frecce indicano in che modo ogni array è associato a diverse istruzioni MB_MASTER.



In tutti gli array o le strutture il primo elemento è sempre costituito dalla prima origine o destinazione di una lettura o scrittura Modbus. Tutte le situazioni possibili sotto illustrate si basano sullo schema sopra rappresentato.

Situazione 1: se la prima istruzione MB_MASTER legge 3 parole di dati dell'indirizzo Modbus 40001 in qualsiasi slave Modbus valido, accade quanto segue.

La parola dall'indirizzo 40001 è memorizzata in "Data".Array_1[1].

La parola dall'indirizzo 40002 è memorizzata in "Data".Array_1[2].

La parola dall'indirizzo 40003 è memorizzata in "Data".Array_1[3].

Scenario 2: se la prima istruzione MB_MASTER legge 4 parole di dati dall'indirizzo Modbus 40015 in qualsiasi slave Modbus valido, accade quanto segue.

La parola dell'indirizzo 40015 viene memorizzata in "Data".Array_1[1].

La parola dell'indirizzo 40016 viene memorizzata in "Data".Array_1[2].

La parola dell'indirizzo 40017 viene memorizzata in "Data".Array_1[3].

La parola dell'indirizzo 40018 viene memorizzata in "Data".Array_1[4].

Scenario 3: se la seconda istruzione MB_MASTER legge 2 parole di dati dall'indirizzo Modbus 30033 in qualsiasi slave Modbus valido, accade quanto segue.

La parola dell'indirizzo 30033 viene memorizzata in "Data".Array_2[1].

La parola dell'indirizzo 30034 viene memorizzata in "Data".Array_2[2].

Scenario 4: se la terza istruzione MB_MASTER scrive 4 parole di dati nell'indirizzo Modbus 40050 in qualsiasi slave Modbus valido, accade quanto segue.

La parola di "Data".Array_3[1] viene scritta nell'indirizzo Modbus 40050.

La parola di "Data".Array_3[2] viene scritta nell'indirizzo Modbus 40051.

La parola di "Data".Array_3[3] viene scritta nell'indirizzo Modbus 40052.

La parola di "Data".Array_3[4] viene scritta nell'indirizzo Modbus 40053.

Scenario 5: se la terza istruzione MB_MASTER scrive 3 parole di dati nell'indirizzo Modbus 40001 in qualsiasi slave Modbus valido, accade quanto segue.

La parola di "Data".Array_3[1] viene scritta nell'indirizzo Modbus 40001.

La parola di "Data".Array_3[2] viene scritta nell'indirizzo Modbus 40002.

La parola di "Data".Array_3[3] viene scritta nell'indirizzo Modbus 40003.

Scenario 6: se la quarta istruzione MB_MASTER utilizza un Modo 11 (recupero del conteggio dei messaggi validi) da uno qualsiasi degli slave Modbus validi, accade quanto segue.

Il numero di parole viene memorizzato in "Data".Array_4[1].

Esempi di lettura e scrittura di bit tramite indirizzi di parola come ingresso per DATA_PTR

Tabella 6- 1 Scenario 7: leggi 4 bit di uscita a partire dall'indirizzo Modbus 00001

Valori di ingresso di MB_MASTER		Valori dello slave Modbus	
MB_ADDR	27 (esempio di slave)	00001	ON
MODE	0 (lettura)	00002	ON
DATA_ADDR	00001 (uscite)	00003	OFF
DATA_LEN	4	00004	ON
DATA_PTR	"Data".Array_4	00005	ON
		00006	OFF
		00007	ON
		00008	OFF

Valori di "Data".Array_4[1] dopo la richiesta Modbus	
Byte MS (più significativo)	Byte LS (meno significativo)
xxxx-1011	xxxx-xxxx
x indica che i dati non sono cambiati	

Tabella 6- 2 Scenario 8: leggi 12 bit di uscita a partire dall'indirizzo Modbus 00003

Valori di ingresso di MB_MASTER		Valori dello slave Modbus			
MB_ADDR	27 (esempio di slave)	00001	ON	00010	ON
MODE	0 (lettura)	00002	ON	00011	OFF
DATA_ADDR	00003 (uscite)	00003	OFF	00012	OFF
DATA_LEN	12	00004	ON	00013	ON
DATA_PTR	"Data".Array_4	00005	ON	00014	OFF
		00006	OFF	00015	ON
		00007	ON	00016	ON
		00008	ON	00017	OFF
		00009	OFF	00018	ON

Valori di "Data".Array_4[1] dopo la richiesta Modbus	
Byte MS	Byte LS
1011-0110	xxxx-0100-
x indica che i dati non sono cambiati	

Tabella 6- 3 Scenario 9: scrivi 5 bit di uscita a partire dall'indirizzo Modbus 00001

Valori di ingresso di MB_MASTER		Uscite dello slave prima		Uscite dello slave dopo
MB_ADDR	27 (esempio di slave)	00001	ON	OFF
MODE	1 (scrittura)	00002	ON	ON
DATA_ADDR	00001 (uscite)	00003	OFF	ON
DATA_LEN	5	00004	ON	OFF
DATA_PTR	"Data".Array_4	00005	ON	ON
		00006	OFF	Invariato
		00007	ON	Invariato
		00008	ON	Invariato
		00009	OFF	Invariato

Valori di "Data".Array_4[1] per la richiesta di scrittura Modbus	
Byte MS	Byte LS
xxx1-0110	xxxxx-xxxx
x indica che i dati non vengono utilizzati nella richiesta Modbus	

Tabella 6- 4 Scenario 10: leggi 22 bit di uscita a partire dall'indirizzo Modbus 00003

Valori di ingresso di MB_MASTER		Valori dello slave Modbus			
MB_ADDR	27 (esempio di slave)	00001	ON	00014	ON
MODE	0 (lettura)	00002	ON	00015	OFF
DATA_ADDR	00003 (uscite)	00003	OFF	00016	ON
DATA_LEN	22	00004	ON	00017	ON
DATA_PTR	"Data".Array_4	00005	ON	00018	OFF
		00006	OFF	00019	ON
		00007	ON	00020	ON
		00008	ON	00021	OFF
		00009	ON	00022	ON
		00010	OFF	00023	ON
		00011	OFF	00024	OFF
		00012	ON	00025	OFF
		00013	OFF	00026	ON

Valori di "Data".Array_4[1] dopo la richiesta Modbus	
Byte MS	Byte LS
0111-0110	0110-1010

Valori di "Data".Array_4[2] dopo la richiesta Modbus	
Byte MS	Byte LS
xx01-1011	xxxx-xxxx
x indica che i dati non sono cambiati	

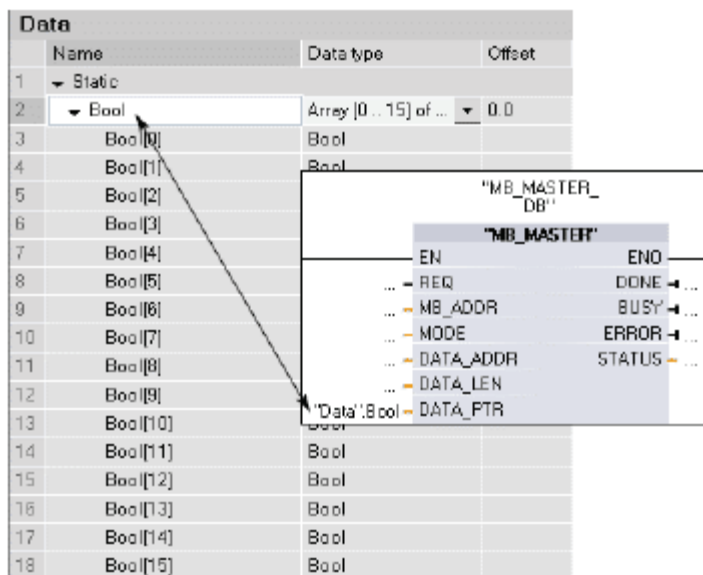
Esempi di lettura e scrittura di bit tramite indirizzi BOOL come ingresso per DATA_PTR

Nonostante le letture e le scritture Modbus negli indirizzi di bit possano essere gestiti tramite indirizzi di parola, le aree DATA_PTR possono anche essere configurate come tipi di dati booleani, strutture o array in modo da garantire una corrispondenza diretta 1 a 1 tra il primo bit letto o scritto mediante un'istruzione MB_MASTER.

Se si utilizzano array o strutture booleane si consiglia di utilizzare una dimensione di dati che sia un multiplo di 8 bit (con limiti di byte). Ad esempio, se si crea un array booleano di 10 bit il software STEP 7 Basic assegna loro 16 bit (2 byte) del blocco dati globale. All'interno del blocco dati questi bit vengono memorizzati come byte1 [xxxx xxxx] byte2 [---- --xx] dove la x indica gli indirizzi di dati accessibili e – quelli inaccessibili. Sono consentite richieste Modbus fino a una lunghezza di 16 bit ma i 6 bit superiori vengono inseriti negli indirizzi di memoria del byte 2 che non vengono indirizzati e non sono accessibili dal programma.

È possibile creare aree booleane come array di valori booleani o struttura di variabili booleane. Entrambi i sistemi funzionano allo stesso modo e si differenziano solo per il metodo di creazione e di accesso da parte del programma.

La vista dell'editor dei blocchi dati globali sotto rappresentata mostra un array singolo costituito da 16 valori booleani creato a base 0. L'array potrebbe essere stato creato anche come array a base 1. Le frecce indicano in che modo l'array è associato all'istruzione MB_MASTER.



Gli scenari 11 e 12 mostrano la corrispondenza degli indirizzi Modbus con gli indirizzi degli array booleani.

Tabella 6- 5 Scenario 11: scrivi 5 bit di uscita a partire dall'indirizzo Modbus 00001

Valori di ingresso di MB_MASTER		Uscite dello slave prima		Dati di DATA_PTR	Uscite dello slave dopo
MB_ADDR	27 (esempio di slave)	00001	ON	"Data".Bool[0]=falso	OFF
MODE	1 (scrittura)	00002	ON	"Data".Bool[1]=vero	ON
DATA_ADDR	00001 (uscite)	00003	OFF	"Data".Bool[2]=vero	ON
DATA_LEN	5	00004	ON	"Data".Bool[3]=falso	OFF
DATA_PTR	"Data".Bool	00005	ON	"Data".Bool[4]=falso	OFF
		00006	OFF		Invariato
		00007	ON		Invariato
		00008	OFF		Invariato

Tabella 6- 6 Scenario 12: leggi 15 bit di uscita a partire dall'indirizzo Modbus 00004

Valori di ingresso di MB_MASTER		Valore dello slave Modbus		Dati di DATA_PTR dopo
MB_ADDR	27 (esempio di slave)	00001	ON	
MODE	0 (lettura)	00002	ON	
DATA_ADDR	00003 (uscite)	00003	OFF	"Data".Bool[0]=falso
DATA_LEN	15	00004	ON	"Data".Bool[1]=vero
DATA_PTR	"Data".Bool	00005	ON	"Data".Bool[2]=vero
		00006	OFF	"Data".Bool[3]=falso
		00007	ON	"Data".Bool[4]=vero
		00008	ON	"Data".Bool[5]=vero
		00009	ON	"Data".Bool[6]=vero
		00010	OFF	"Data".Bool[7]=falso
		00011	OFF	"Data".Bool[8]=falso
		00012	ON	"Data".Bool[9]=vero
		00013	OFF	"Data".Bool[10]=falso
		00014	ON	"Data".Bool[11]=vero
		00015	OFF	"Data".Bool[12]=falso
		00016	ON	"Data".Bool[13]=vero
		00017	ON	"Data".Bool[14]=vero
		00018	OFF	
		00019	ON	

Codici delle condizioni di errore

Valore STATUS (W#16#....)	Descrizione
0000	Nessun errore
80C8	Il timeout di risposta specificato (vedere RCVTIME o MSGTIME) è 0.

Valore STATUS (W#16#....)	Descrizione
80D1	Il ricevitore ha inviato una richiesta di controllo del flusso per sospendere una trasmissione attiva e non ha più ristabilito la trasmissione durante il tempo di attesa specificato. Questo errore viene generato anche durante il controllo di flusso hardware se il ricevitore non invia un CTS entro il tempo di attesa specificato.
80D2	La richiesta di trasmissione è stata interrotta perché non è stato ricevuto alcun segnale DSR dal DCE.
80E0	Il messaggio è stato concluso perché il buffer di ricezione è pieno.
80E1	Il messaggio è stato concluso a causa di un errore di parità.
80E2	Il messaggio è stato concluso a causa di un errore di framing.
80E3	Il messaggio è stato concluso a causa di un errore di overrun.
80E4	Il messaggio è stato concluso perché la lunghezza specificata supera la dimensione complessiva del buffer.
8180	Valore di ID della porta non valido
8186	Indirizzo della stazione Modbus non valido
8188	Valore MODE o valore di scrittura per la lettura dell'area degli indirizzi dello slave non valido.
8189	Valore degli indirizzi dei dati non valido
818A	Valore di lunghezza dei dati non valido
818B	Puntatore non valido all'origine/sorgente dei dati: dimensione non corretta
818C	Puntatore ad un DB DATA_PTR di tipo sicuro (deve essere un DB di tipo classico)
8200	La porta è occupata dall'elaborazione di una richiesta di trasmissione

6.3.2.3 MB_SLAVE

L'istruzione MB_SLAVE consente al programma di comunicare come uno slave Modbus utilizzando una porta del modulo punto a punto (PtP) CM 1241 RS485 o CM 1241 RS232. Un master Modbus RTU può inviare una richiesta alla quale il programma risponde eseguendo MB_SLAVE.

Quando si inserisce l'istruzione MB_SLAVE nel programma è necessario assegnarle un blocco dati di istanza univoco. Il nome del DB viene utilizzato quando si specifica il parametro MB_DB dell'istruzione MB_COMM_LOAD.

I codici delle funzioni di comunicazione Modbus (1, 2, 4, 5 e 15) possono leggere e scrivere direttamente bit e parole nell'immagine di processo degli ingressi e delle uscite del PLC. La seguente tabella descrive l'associazione degli indirizzi modbus all'immagine di processo nella CPU.

Funzioni Modbus MB_SLAVE						S7-1200	
Codici	Funzione	Area di dati	Campo degli indirizzi			Area di dati	Indirizzo della CPU
01	Leggi bit	Uscita	1	...	8192	Immagine di processo delle uscite	Q0.0 ... Q1023.7
02	Leggi bit	Ingresso	10001	...	18192	Immagine di processo degli ingressi	I0.0 ... I1023.7
04	Leggi parole	Ingresso	30001	...	30512	Immagine di processo degli ingressi	IW0 ... IW1022
05	Scrivi bit	Uscita	1	...	8192	Immagine di processo delle uscite	Q0.0 ... Q1023.7
15	Scrivi bit	Uscita	1	...	8192	Immagine di processo delle uscite	Q0.0 ... Q1023.7

I codici delle funzioni di comunicazione Modbus (3, 6, 16) utilizzano un blocco dati specifico e univoco del registro di mantenimento Modbus, che deve essere creato per poter specificare il parametro MB_HOLD_REG dell'istruzione MB_SLAVE. La seguente tabella descrive l'associazione dei registri di mantenimento modbus all'indirizzo del DB MB_HOLD_REG nella CPU.

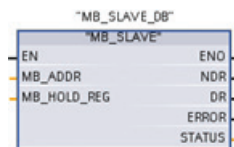
Funzioni Modbus MB_SLAVE				S7-1200	
Codici	Funzione	Area di dati	Campo degli indirizzi	Area di dati DB della CPU	Indirizzo DB della CPU
03	Leggi parole	Registro di mantenimento	Da 40001 a 49999	MB_HOLD_REG	Parole da 1 a 9999
			Da 400001 a 465535		Parole da 1 a 65535
06	Scrivi parola	Registro di mantenimento	Da 40001 a 49999	MB_HOLD_REG	Parole da 1 a 9999
			Da 400001 a 465535		Parole da 1 a 65535
16	Scrivi parole	Registro di mantenimento	Da 40001 a 49999	MB_HOLD_REG	Parole da 1 a 9999
			Da 400001 a 465535		Parole da 1 a 65535

La seguente tabella illustra le funzioni di diagnostica Modbus.

Funzioni di diagnostica Modbus di MB_SLAVE per S7-1200		
Codici	Sotto-funzione	Descrizione
08	0000H	Restituisce il test di eco dei dati: MB_SLAVE restituisce a un master Modbus l'eco di una parola di dati ricevuta.
08	000AH	Resetta il contatore degli eventi di comunicazione: MB_SLAVE resetta il contatore degli eventi di comunicazione utilizzato per la funzione Modbus 11.
11		Legge il contatore degli eventi di comunicazione: MB_SLAVE utilizza un contatore interno degli eventi di comunicazione per registrare il numero di richieste di lettura e scrittura Modbus eseguite correttamente che vengono inviate allo slave Modbus. Il valore del contatore non viene incrementato in seguito alle richieste della funzione 8, della funzione 11 o di trasmissione e nemmeno in seguito alle richieste che si concludono con un errore di comunicazione (ad esempio errori di parità o di CRC).

MB_SLAVE supporta le richieste di trasmissione di scrittura provenienti da qualsiasi master Modbus, purché prevedano l'accesso a indirizzi validi.

MB_SLAVE non invia al master Modbus alcuna risposta alle sue richieste di trasmissione, indipendentemente dalla loro validità.

KOP**FUP**

Parametro	Tipo di parametro	Tipo di dati	Descrizione
MB_ADDR	IN	USINT	Indirizzo Modbus RTU (da 1 a 247): indirizzo di stazione dello slave Modbus.
MB_HOLD_REG	IN	VARIANT	Puntatore al DB di registro di mantenimento Modbus. Il DB di registro di mantenimento deve essere un DB globale di tipo classico. Vedere la nota relativa a MB_HOLD_REG riportata più sotto.
NDR	OUT	BOOL	Nuovi dati disponibili: 0 – Nessun nuovo dato 1 – Indica che il master Modbus ha scritto nuovi dati
DR	OUT	BOOL	Dati letti: 0 – Nessun dato letto 1 – Indica che il master Modbus ha letto dei dati
ERROR	OUT	BOOL	Errore: 0 – Nessun errore rilevato 1 – Indica che è stato rilevato un errore e che il relativo codice indicato nel parametro STATUS è valido.
STATUS	OUT	WORD	Codice di errore

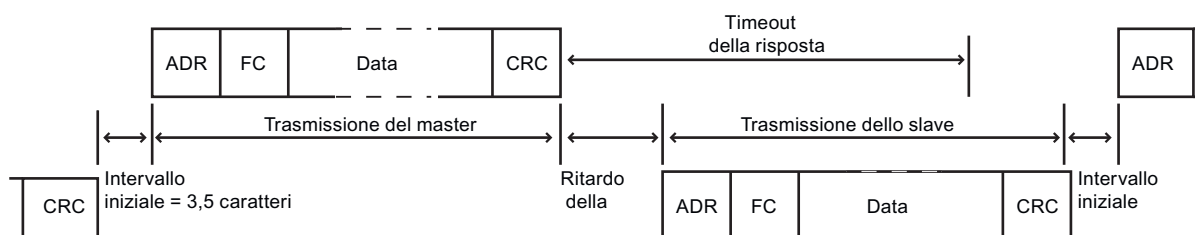
Regole di comunicazione dello slave Modbus

- Perché un'istruzione MB_SLAVE possa comunicare con una porta è necessario che questa sia stata configurata eseguendo MB_COMM_LOAD.
- La porta che deve rispondere come slave a un master Modbus non può essere utilizzata da MB_MASTER. Con una determinata porta è consentito utilizzare una sola istanza dell'esecuzione di MB_SLAVE.
- Le istruzioni Modbus non utilizzano eventi di allarme per comandare il processo di comunicazione. Per il comando della comunicazione il programma deve interrogare l'istruzione MB_SLAVE e richiedere le condizioni complete di trasmissione e ricezione.

- MB_SLAVE deve essere eseguita periodicamente ad una velocità che le consenta di reagire tempestivamente alle richieste provenienti da un master Modbus.
- La si deve richiamare in tutti i cicli di scansione da un OB di ciclo.

Funzionamento

L'istruzione MB_SLAVE deve essere eseguita periodicamente, in modo da ricevere tutte le richieste dal master Modbus e rispondere in modo adeguato. La frequenza di esecuzione dipende dal periodo di timeout della risposta del master Modbus come illustrato nel seguente schema.



Il periodo di timeout della risposta indica per quanto tempo un master Modbus resta in attesa che lo slave Modbus inizi ad inviare una risposta. Questo periodo di tempo non viene definito dal protocollo Modbus ma è un parametro di ciascun master Modbus. La frequenza di esecuzione (il tempo che intercorre tra un'esecuzione e l'altra) di MB_SLAVE deve basarsi su parametri specifici del master Modbus utilizzato. Questa istruzione dovrebbe essere eseguita almeno due volte entro il periodo di timeout della risposta del master Modbus.

Esempi di parametro MB_HOLD_REG

MB_HOLD_REG è un puntatore al blocco dati di registro di mantenimento Modbus. Questo DB viene utilizzato per mantenere i valori dei dati ai quali un master Modbus è autorizzato ad accedere (in lettura o in scrittura). Per poterlo utilizzare con l'istruzione MB_SLAVE è quindi necessario creare il blocco dati e definire la struttura del tipo di dati ai quali è possibile accedere in lettura e in scrittura.

Nota

Il blocco dati del registro di mantenimento Modbus deve far riferimento a un blocco dati globale per il quale è stato attivato l'attributo Indirizzabile soltanto simbolicamente.

Quando si aggiunge un nuovo blocco dati per creare un DB di questo tipo è necessario deselezionare la casella "Indirizzabile soltanto simbolicamente".

I registri di mantenimento possono utilizzare le seguenti strutture di dati:

- Array di parole standard
- Struttura costituita da parole definite da un nome
- Struttura complessa definita da nomi

Gli esempi di programmazione riportati di seguito mostrano come utilizzare il parametro MB_HOLD_REG per gestire queste strutture.

Esempio 1 - Array di parole standard

Il registro di mantenimento illustrato in questo esempio è formato da un array di parole. I tipi di dati assegnati possono essere modificati in modo che utilizzino altri tipi di formati in parola (INT e UINT).

Vantaggi:

- I registri di mantenimento con questo tipo di struttura sono estremamente semplici e rapidi da creare.
- La logica del programma per l'accesso agli elementi di dati viene semplificata.

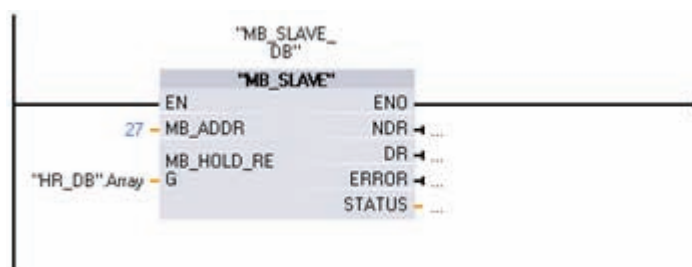
Svantaggi:

- Sebbene a livello di programma sia possibile fare riferimento a ciascun elemento dell'array mediante il relativo nome simbolico (da "HR_DB"."Array"[1] a "HR_DB"."Array"[10]), i nomi non descrivono la funzione interna dei dati.
- L'array può essere costituito da un solo tipo di dati. In un programma utente con un controllo rigido del tipo potrebbe quindi essere necessario convertire i tipi di dati.

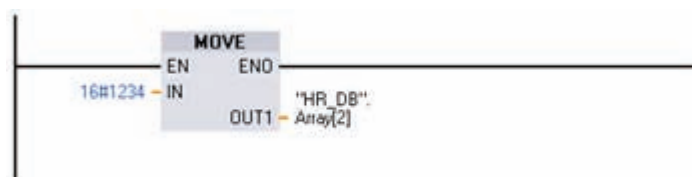
Nell'editor di blocchi dati una struttura costituita da array di parole verrebbe rappresentata come illustrato di seguito.

HR_DB							
Name	Data type	Offset	Default value	Initial value	Retain	Comment	
1 Static							
2 Array	Array [1..10] of W.	0.0			☐	40001 to 40010	
3 Array[1]	Word		W#16#0000	W#16#0000	☐		
4 Array[2]	Word		W#16#0000	W#16#0000	☐		
5 Array[3]	Word		W#16#0000	W#16#0000	☐		
6 Array[4]	Word		W#16#0000	W#16#0000	☐		
7 Array[5]	Word		W#16#0000	W#16#0000	☐		
8 Array[6]	Word		W#16#0000	W#16#0000	☐		
9 Array[7]	Word		W#16#0000	W#16#0000	☐		
10 Array[8]	Word		W#16#0000	W#16#0000	☐		
11 Array[9]	Word		W#16#0000	W#16#0000	☐		
12 Array[10]	Word		W#16#0000	W#16#0000	☐		

La seguente figura mostra in che modo l'array verrebbe assegnato all'ingresso MB_HOLD_REG di un'istruzione MB_SLAVE.



Ogni elemento dell'array è accessibile tramite un nome simbolico come illustrato di seguito. Nell'esempio un nuovo valore viene trasferito nel secondo elemento dell'array corrispondente all'indirizzo Modbus 40002.



Ogni parola dell'array definita nel blocco dati fornisce all'istruzione MB_SLAVE gli indirizzi del registro di mantenimento Modbus. Poiché in questa istanza l'array contiene solo 10 elementi, anche gli indirizzi del registro di mantenimento Modbus utilizzabili dall'istruzione MB_SLAVE e ai quali il master Modbus può accedere sono solamente 10.

Qui di seguito è illustrata la correlazione tra i nomi degli elementi dell'array e gli indirizzi Modbus.

"HR_DB".Array[1]	Indirizzo Modbus 40001
" HR_DB ". Array[2]	Indirizzo Modbus 40002
" HR_DB ". Array[3]	Indirizzo Modbus 40003
...	...
" HR_DB ". Array[9]	Indirizzo Modbus 40009
" HR_DB ".Array [10]	Indirizzo Modbus 40010

Esempio 2 - Struttura costituita da parole definite da un nome

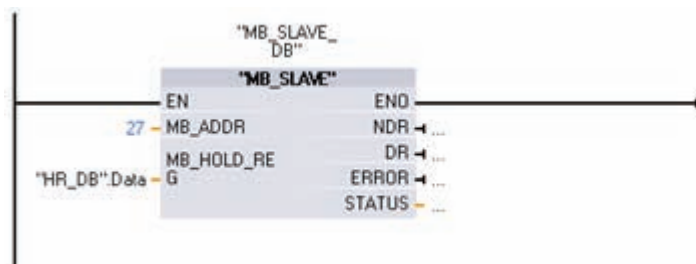
Il registro di mantenimento illustrato in questo esempio è costituito da una serie di parole con nomi simbolici descrittivi.

- | | |
|-------------------|--|
| Vantaggi: | <ul style="list-style-type: none">• Ogni elemento della struttura ha un nome descrittivo al quale è assegnato un tipo specifico di dati. |
| Svantaggi: | <ul style="list-style-type: none">• La creazione di questo tipo di struttura richiede più tempo rispetto agli array di parole standard.• Se utilizzati nel programma utente gli elementi richiedono un ulteriore indirizzamento simbolico. Mentre il primo elemento dell'array semplice è indirizzato con "HR_DB".Array[0], il primo elemento di questo tipo viene indirizzato con "HR_DB".Data.Temp_1. |

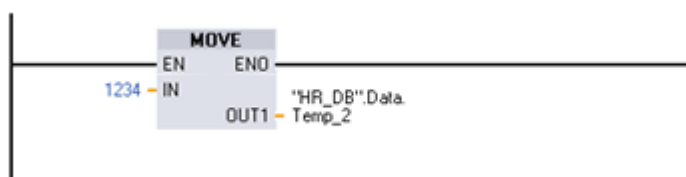
Nell'editor di blocchi dati una struttura costituita da parole definite da nomi verrebbe rappresentata come illustrato di seguito. Ogni elemento (WORD, UINT o INT) ha un nome univoco.

HR_DB					
	Name	Data type	Offset	Initial value	Retain
1	Static				
2	▼ Data	Struct	▼ 0.0		☐
3	Temp_1	Int	0.0	0	☐
4	Temp_2	Int	2.0	0	☐
5	Temp_3	Int	4.0	0	☐
6	Good_Count	UInt	6.0	0	☐
7	Bad_Count	UInt	8.0	0	☐
8	Rework_Count	UInt	10.0	0	☐
9	Line_Stops	UInt	12.0	0	☐
10	Avg_Time	UInt	14.0	0	☐
11	Code_1	Word	16.0	0	☐
12	Code_2	Word	18.0	0	☐

La figura che segue mostra in che modo questa struttura di dati verrebbe assegnata all'ingresso MB_HOLD_REG di un'istruzione MB_SLAVE in un programma.



Ogni elemento dell'array è accessibile tramite il suo nome simbolico, come illustrato di seguito. Nell'esempio un nuovo valore viene trasferito nel secondo elemento dell'array corrispondente all'indirizzo Modbus 40002.



Di seguito è illustrata la correlazione tra i nomi degli elementi di dati e gli indirizzi Modbus.

"HR_DB".Data.Temp_1	Indirizzo Modbus 40001
"HR_DB".Data.Temp_2	Indirizzo Modbus 40002
"HR_DB".Data.Temp_3	Indirizzo Modbus 40003
"HR_DB".Data.Good_Count	Indirizzo Modbus 40004
"HR_DB".Data.Bad_Count	Indirizzo Modbus 40005
"HR_DB".Data.Rework_Count	Indirizzo Modbus 40006
"HR_DB".Data.Line_Stops	Indirizzo Modbus 40007
"HR_DB".Data.Avg_Time	Indirizzo Modbus 40008
"HR_DB".Data.Code_1	Indirizzo Modbus 40009
"HR_DB".Data.Code_2	Indirizzo Modbus 40010

Esempio 3 - Struttura complessa associata a nomi

Il registro di mantenimento illustrato in questo esempio è una serie di tipi di dati misti definiti da nomi simbolici descrittivi.

Vantaggi:

- Ogni elemento della struttura ha un nome descrittivo al quale è assegnato un tipo specifico di dati.
- Permette il trasferimento diretto dei tipi di dati non basati sulle parole.

Svantaggi:

- La creazione di questo tipo di struttura richiede più tempo rispetto agli array di parole standard.
- Il master Modbus deve essere configurato in modo da accettare i dati che riceverà dallo slave Modbus. Come illustrato nella figura che segue Temp_1 è un valore reale di 4 byte. Il master ricevente deve essere in grado di ricomporre le 2 parole ricevute ricostruendo il valore di numero reale previsto.
- Gli elementi richiedono un ulteriore indirizzamento simbolico all'interno del programma. Mentre il primo elemento dell'array semplice è indirizzato con "HR_DB".Array[0], il primo elemento di questo tipo viene indirizzato con "HR_DB".Data.Temp_1.

Nell'editor di blocchi dati una struttura complessa associata a nomi verrebbe rappresentata come illustrato di seguito. Ogni elemento ha un nome univoco con diversi formati e tipi di dati.

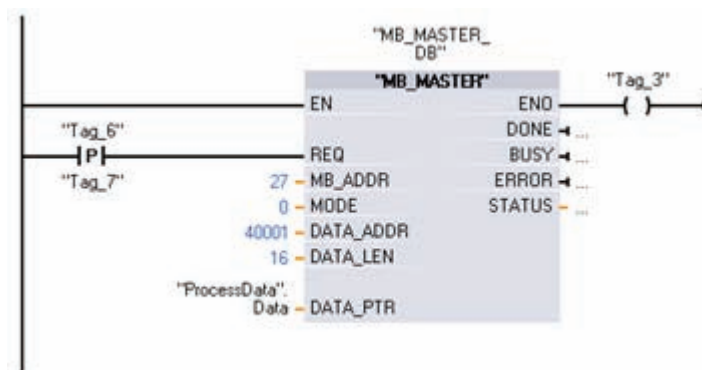
HR_DB						
	Name	Data type	Offset	Initial value	Retain	Comment
1	Static					
2	Data	Struct	0.0		☐	Modbus addresses 40001 to 40016
3	Temp_1	Real	0.0	0.0	☐	40001 and 40002
4	Temp_2	Real	4.0	0.0	☐	40003 and 40004
5	Good_Count	UDInt	8.0	0	☐	40005 and 40006
6	Bad_Count	UDInt	12.0	0	☐	40007 and 40008
7	Rework_Count	UDInt	16.0	0	☐	40009 and 40010
8	Line_Stops	Int	20.0	0	☐	40011
9	Avg_Time	Int	22.0	0	☐	40012
10	Long_Code	DWord	24.0	0	☐	40013 and 40014
11	Code_1	Word	28.0	0	☐	40015
12	Code_2	Word	30.0	0	☐	40016

Di seguito è illustrata la correlazione tra i nomi degli elementi di dati e gli indirizzi Modbus.

"HR_DB".Data.Temp_1	Indirizzi Modbus 40001 e 40002
"HR_DB".Data.Temp_2	Indirizzi Modbus 40003 e 40004
"HR_DB".Data.Good_Count	Indirizzi Modbus 40005 e 40006
"HR_DB".Data.Bad_Count	Indirizzi Modbus 40007 e 40008
"HR_DB".Data.Rework_Count	Indirizzi Modbus 40009 e 40010
"HR_DB".Data.Line_Stops	Indirizzo Modbus 400011
"HR_DB".Data.Avg_Time	Indirizzo Modbus 400012
"HR_DB".Data.Long_Code	Indirizzi Modbus 40013 e 40014
"HR_DB".Data.Code_1	Indirizzo Modbus 40015
"HR_DB".Data.Code_2	Indirizzo Modbus 40016

Un'altra CPU S7-1200 con funzione di master Modbus può utilizzare l'istruzione MB_Master e una struttura di dati identica per ricevere il blocco dati dalla CPU S7-1200 che funge da

slave Modbus. Questa istruzione fa sì che tutte e 16 le parole di dati vengano copiate direttamente dal blocco dati HR_DB dello slave nel blocco dati ProcessData del master, come illustrato di seguito.



ProcessData						
Name	Data type	Offset	Initial value	Retain	Comment	
1 Static						
2 Data	Struct	0.0		☐	Modbus addresses 40001 to 40016	
3 Temp_1	Real	0.0	0.0	☐	40001 and 40002	
4 Temp_2	Real	4.0	0.0	☐	40003 and 40004	
5 Good_Count	UDInt	8.0	0	☐	40005 and 40006	
6 Bad_Count	UDInt	12.0	0	☐	40007 and 40008	
7 Rework_Count	UDInt	16.0	0	☐	40009 and 40010	
8 Line_Stops	Int	20.0	0	☐	40011	
9 Avg_Time	Int	22.0	0	☐	40012	
10 Long_Code	DWord	24.0	0	☐	40013 and 40014	
11 Code_1	Word	28.0	0	☐	40015	
12 Code_2	Word	30.0	0	☐	40016	

È possibile trasferire la stessa struttura o strutture diverse da più slave Modbus utilizzando una serie di indirizzi di blocchi dati Data_PTR del master Modbus.

Codici delle condizioni di errore

Valore STATUS (W#16#....)	Descrizione
80C8	Il timeout di risposta specificato (fare riferimento a RCVTIME o MSGTIME) è 0
80D1	Il ricevitore ha inviato una richiesta di controllo del flusso per sospendere una trasmissione attiva e non ha più ristabilito la trasmissione durante il tempo di attesa specificato. Questo errore viene generato anche durante il controllo di flusso hardware se il ricevitore non invia un CTS entro il tempo di attesa specificato.
80D2	La richiesta di trasmissione è stata interrotta perché non è stato ricevuto alcun segnale DSR dal DCE
80E0	Il messaggio è stato concluso perché il buffer di ricezione è pieno
80E1	Il messaggio è stato concluso a causa di un errore di parità
80E2	Il messaggio è stato concluso a causa di un errore di framing
80E3	Il messaggio è stato concluso a causa di un errore di overrun
80E4	Il messaggio è stato concluso perché la lunghezza specificata supera la dimensione complessiva del buffer
8180	Valore di ID della porta non valido

Valore STATUS (W#16#....)	Descrizione	
8186	Indirizzo della stazione Modbus non valido	
8187	Puntatore al DB MB_HOLD_REG non valido	
818C	Puntatore ad un DB MB_HOLD_REG di tipo sicuro (deve essere un DB di tipo classico)	
	Codice di risposta inviato al master Modbus (B#16#..)	
8380	Nessuna risposta	Errore CRC
8381	01	Codice della funzione non supportato
8382	Nessuna risposta	Lunghezza dei dati errata
8383	02	Indirizzo dei dati errato
8384	03	Valore dei dati errato
8385	03	Valore del codice di diagnostica dei dati non supportato (codice funzione 08)

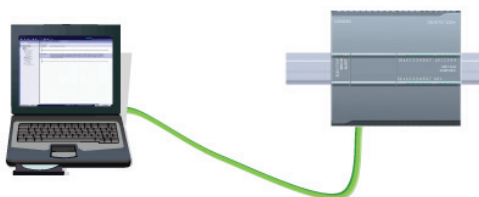
PROFINET

La CPU S7-1200 ha una porta PROFINET integrata che supporta gli standard di comunicazione basati su Ethernet e TCP/IP. La CPU S7-1200 supporta i seguenti protocolli applicativi:

- Protocollo di controllo TCP (Transport Control Protocol)
- ISO on TCP (RFC 1006)

La CPU S7-1200 può comunicare con altre CPU S7-1200, con il dispositivo di programmazione STEP 7 Basic, con i dispositivi HMI e con i dispositivi non Siemens mediante i protocolli di comunicazione TCP standard. La comunicazione tramite PROFINET può avvenire in due modi:

- Collegamento diretto: utilizzare la comunicazione diretta quando si collega un dispositivo di programmazione, un HMI o un'altra CPU ad una sola CPU.
- Collegamento di rete: utilizzare la comunicazione di rete quando si collegano tra loro più di due dispositivi (ad esempio CPU, HMI, dispositivi di programmazione e dispositivi non Siemens).



Collegamento diretto: dispositivo di programmazione collegato alla CPU S7-1200



Collegamento diretto: HMI collegato alla CPU S7-1200



Collegamento diretto: una CPU S7-1200 collegata ad un'altra CPU S7-1200



Collegamento di rete:
più di due dispositivi
collegati tra loro con
uno switch Ethernet
CSM1277 ①

Per il collegamento diretto tra un dispositivo di programmazione o un HMI e una CPU non è necessario uno switch Ethernet, che è invece indispensabile per una rete con più di due CPU o dispositivi HMI. Per collegare le CPU e i dispositivi HMI è possibile utilizzare lo switch Ethernet a 4 porte CSM1277 Siemens montato su telaio. La porta PROFINET della CPU S7-1200 non contiene un dispositivo di commutazione Ethernet.

Numero massimo di collegamenti per la porta PROFINET

La porta PROFINET della CPU supporta i seguenti collegamenti di comunicazione paralleli.

- 3 collegamenti per HMI per la comunicazione con la CPU
- 1 collegamento per il dispositivo di programmazione (PG) per la comunicazione con la CPU
- 8 collegamenti per la comunicazione con il programma S7-1200 mediante le istruzioni del blocco T (TSEND_C, TRCV_C, TCON, TDISCON, TSEN, TRCV)
- 3 collegamenti per una CPU S7-1200 passiva che comunica con una CPU S7 attiva
 - La CPU S7 attiva utilizza le istruzioni GET e PUT (S7-300 e S7-400) o ETHx_XFER (S7-200).
 - Un collegamento attivo con S7-1200 è possibile solo mediante le istruzioni del blocco T.

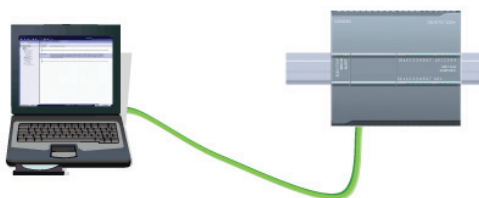
TSAP o numeri di porta limitati per la comunicazione ISO e TPC attiva

Se si utilizza l'istruzione "TCON" per impostare e stabilire un collegamento passivo, i seguenti indirizzi della porta sono limitati e non dovrebbero essere utilizzati:

- TSAP ISO (passivo): 01.00, 01.01, 02.00, 02.01, 03.00, 03.01
- Porta TCP (passiva): 5001, 102, 123, 20, 21, 25, 34962, 34963, 34964, 80

7.1 Comunicazione con un dispositivo di programmazione

Una CPU può comunicare con un dispositivo di programmazione STEP 7 Basic collegato in rete.



Per configurare la comunicazione tra una CPU e un dispositivo di programmazione è necessario tener conto di quanto segue:

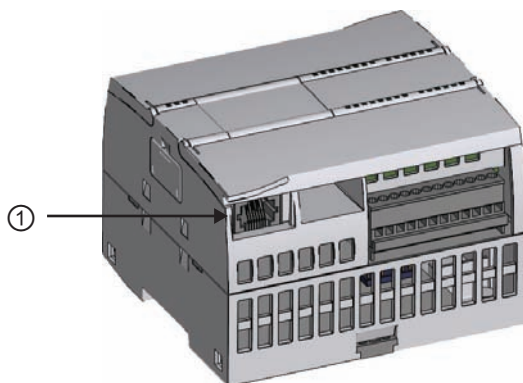
- Configurazione/impostazione: è necessario effettuare la configurazione hardware.
- Per la comunicazione diretta tra due dispositivi non è necessario utilizzare uno switch Ethernet, che è invece indispensabile se si collegano in rete più di due dispositivi.

7.1.1 Realizzazione del collegamento hardware per la comunicazione

Le interfacce PROFINET realizzano i collegamenti fisici tra un dispositivo di programmazione e una CPU. Poiché la CPU dispone della funzionalità Autocrossover, per l'interfaccia è possibile utilizzare indifferentemente un cavo Ethernet di tipo standard o crossover. Il collegamento diretto di un dispositivo di programmazione a una CPU non richiede uno switch Ethernet.

Per realizzare un collegamento hardware tra un dispositivo di programmazione e una CPU procedere nel seguente modo:

1. Installare la CPU (Pagina 29).
2. Innestare il cavo Ethernet nella porta PROFINET sotto illustrata.
3. Collegare il cavo Ethernet al dispositivo di programmazione.

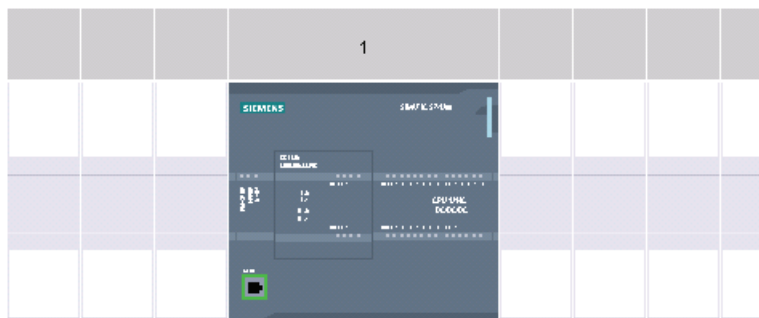


① Porta PROFINET

Il collegamento PROFINET può essere rinforzato con uno scarico della trazione opzionale.

Se è già stato creato un progetto con una CPU, aprirlo nel portale TIA.

In caso contrario creare un progetto e inserire una CPU (Pagina 78) nel telaio di montaggio. Nel progetto sotto illustrato la "Vista dispositivi" del portale TIA visualizza una CPU.



7.1.3.1 Assegnazione degli indirizzi IP ai dispositivi di programmazione e di rete

Se il dispositivo di programmazione utilizza una scheda adattatore on-board collegata alla LAN dell'impianto (ed eventualmente al World Wide Web), l'ID di rete e la maschera di sottorete dell'indirizzo IP della CPU devono essere identici a quelli della scheda adattatore. L'ID di rete è la prima parte dell'indirizzo IP (primi tre ottetti) (ad esempio, **211.154.184**.16) e determina la rete IP sulla quale si sta lavorando. In genere la maschera di sottorete ha il valore **255.255.255.0**, ma poiché il computer fa parte della LAN di un impianto può anche avere valori diversi (ad esempio **255.255.254.0**) in modo da consentire l'impostazione di sottoreti univoche. Quando è associata all'indirizzo IP di un dispositivo in un'operazione matematica di tipo AND, la maschera di sottorete definisce i limiti di una sottorete IP.

Nota

Poiché i dispositivi di programmazione, i dispositivi di rete e i router IP sono collegati al Web e comunicano con il mondo esterno è necessario utilizzare indirizzi IP univoci, in modo da evitare possibili conflitti con altri utenti della rete. Per richiedere l'assegnazione degli indirizzi IP rivolgersi al reparto IT dell'azienda, che conosce a fondo le reti dell'impianto.

Se il dispositivo di programmazione utilizza una scheda adattatore Ethernet-USB collegata a una rete isolata, l'ID di rete e la maschera di sottorete dell'indirizzo IP della CPU devono essere identici a quelli della scheda adattatore. L'ID di rete è la prima parte dell'indirizzo IP (primi tre ottetti) (ad esempio, **211.154.184**.16) e determina la rete IP sulla quale si sta lavorando. In genere la maschera di sottorete ha il valore **255.255.255.0**. Quando è associata all'indirizzo IP di un dispositivo in un'operazione matematica di tipo AND, la maschera di sottorete definisce i limiti di una sottorete IP.

Nota

La scheda adattatore Ethernet-USB può essere utile se non si vuole che la CPU sia collegata alla LAN dell'azienda. Questa soluzione si rivela particolarmente utile durante il collaudo iniziale o i test di messa in servizio.

Scheda adattatore del dispositivo di programmazione	Tipo di rete	Indirizzo IP (Internet Protocol)	Maschera di sottorete
Scheda adattatore on-board	Collegata alla LAN dell'impianto (ed eventualmente al World Wide Web)	L'ID di rete della CPU e quello della scheda adattatore on-board del dispositivo di programmazione devono essere identici. L'ID di rete è la prima parte dell'indirizzo IP (primi due ottetti) (ad esempio 211.154.184.16) e determina la rete IP sulla quale si sta lavorando.	La maschera di sottorete della CPU e quella della scheda adattatore on-board devono essere identiche. In genere la maschera di sottorete ha il valore 255.255.255.0 , ma poiché il computer fa parte della LAN di un impianto può anche avere valori diversi (ad esempio 255.255.254.0) in modo da consentire l'impostazione di sottoreti univoche. Quando è associata all'indirizzo IP di un dispositivo in un'operazione matematica di tipo AND, la maschera di sottorete definisce i limiti di una sottorete IP.
Scheda adattatore da Ethernet a USB	Collegata a una rete isolata	L'ID di rete della CPU e quello della scheda adattatore da Ethernet a USB del dispositivo di programmazione devono essere identici. L'ID di rete è la prima parte dell'indirizzo IP (primi due ottetti) (ad esempio 211.154.184.16) e determina la rete IP sulla quale si sta lavorando.	La maschera di sottorete della CPU e quella della scheda adattatore da Ethernet a USB devono essere identiche. In genere la maschera di sottorete ha il valore 255.255.255.0 . Quando è associata all'indirizzo IP di un dispositivo in un'operazione matematica di tipo AND, la maschera di sottorete definisce i limiti di una sottorete IP.

Assegnazione o controllo dell'indirizzo IP del dispositivo di programmazione utilizzando "Risorse di rete" (sul desktop)

Selezionando le seguenti opzioni di menu è possibile assegnare o controllare l'indirizzo IP del dispositivo di programmazione:

- "Risorse di rete" (con il tasto destro del mouse)
- "Proprietà"
- "Connessione alla rete locale (LAN)" (con il tasto destro del mouse)
- "Proprietà"

Nella finestra di dialogo "Proprietà - Connessione alla rete locale LAN", individuare la voce "Protocollo Internet (TCP/IP)" nel campo "La connessione utilizza i componenti seguenti:". Fare clic su "Protocollo Internet (TCP/IP)" e quindi sul pulsante "Proprietà". Selezionare "Ottieni automaticamente un indirizzo IP (DHCP)" o "Utilizza il seguente indirizzo IP" (per inserire un indirizzo IP statico).

Nota

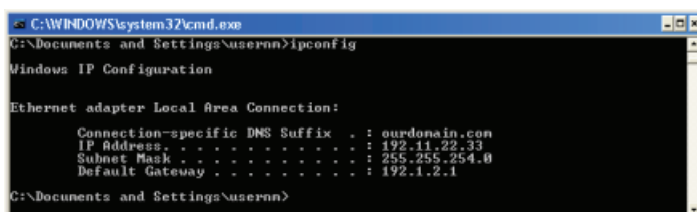
Al momento dell'accensione il Dynamic Host Configuration Protocol (DHCP), un protocollo di configurazione dinamica degli indirizzi, assegna automaticamente al dispositivo di programmazione l'indirizzo IP ricevuto dal server DHCP.

Controllo dell'indirizzo IP del dispositivo di programmazione con i comandi "ipconfig" e "ipconfig /all"

Selezionando le seguenti voci di menu è inoltre possibile controllare l'indirizzo IP del dispositivo di programmazione ed eventualmente quello del router IP (gateway):

- Pulsante "Start" (sul desktop)
- "Esegui"

Nella finestra "Esegui", digitare "cmd" nel campo "Apri" e fare clic sul pulsante "OK". Quindi digitare il comando "ipconfig" nella finestra "C:\WINDOWS\system32\cmd.exe" visualizzata. Di seguito viene illustrato un possibile esempio delle informazioni che si possono ottenere:



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\userna>ipconfig

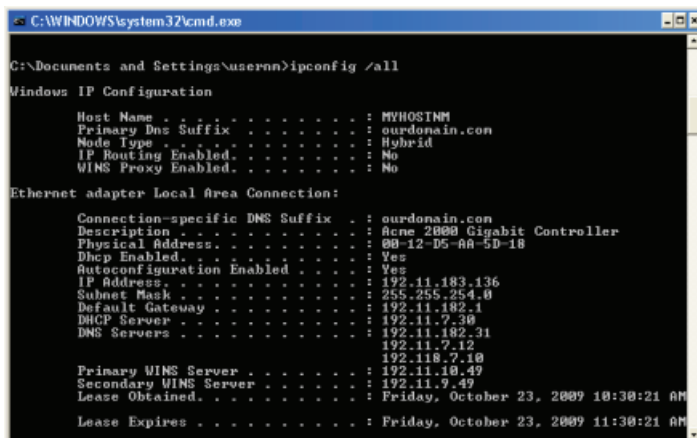
Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : ourdonain.com
    IP Address. . . . . : 192.11.22.33
    Subnet Mask . . . . . : 255.255.254.0
    Default Gateway . . . . . : 192.1.2.1

C:\Documents and Settings\userna>
```

Utilizzando il comando "ipconfig /all" è inoltre possibile ottenere ulteriori informazioni. Nella finestra visualizzata compare il tipo di scheda adattatore utilizzata dal dispositivo di programmazione e l'indirizzo Ethernet (MAC):



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\userna>ipconfig /all

Windows IP Configuration

    Host Name . . . . . : MYHOSTNM
    Primary Dns Suffix . . . . . : ourdonain.com
    Node Type . . . . . : Hybrid
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : ourdonain.com
    Description . . . . . : Acme 2800 Gigabit Controller
    Physical Address. . . . . : 00-12-D5-AA-5D-18
    Dhcp Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IP Address. . . . . : 192.11.183.136
    Subnet Mask . . . . . : 255.255.254.0
    Default Gateway . . . . . : 192.11.182.1
    DHCP Server . . . . . : 192.11.7.30
    DNS Servers . . . . . : 192.11.182.31
                           192.11.7.12
                           192.118.7.10
    Primary WINS Server . . . . . : 192.11.10.49
    Secondary WINS Server . . . . . : 192.11.9.49
    Lease Obtained. . . . . : Friday, October 23, 2009 10:30:21 AM
    Lease Expires . . . . . : Friday, October 23, 2009 11:30:21 AM
```

Assegnazione di un indirizzo IP ad una CPU

Esistono due metodi per assegnare gli indirizzi IP a una CPU:

- Assegnazione dell'indirizzo IP online
- Configurazione di un indirizzo IP nel progetto

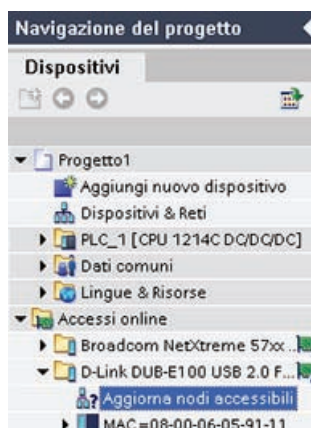
7.1.3.2 Assegnazione dell'indirizzo IP online

Il sistema permette di assegnare un indirizzo IP ad un dispositivo di rete online, una soluzione particolarmente utile durante la configurazione iniziale di un dispositivo.

Per l'assegnazione online di un indirizzo IP procedere come segue:

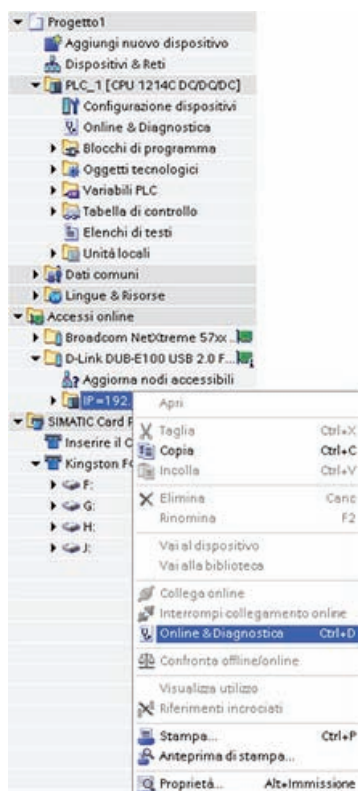
1. Nell'albero del progetto, verificare che non sia stato assegnato un indirizzo IP alla CPU selezionando le seguenti voci di menu:

- "Accesso online"
- <Scheda adattatore per la rete nella quale si trova il dispositivo>
- "Aggiorna nodi accessibili"



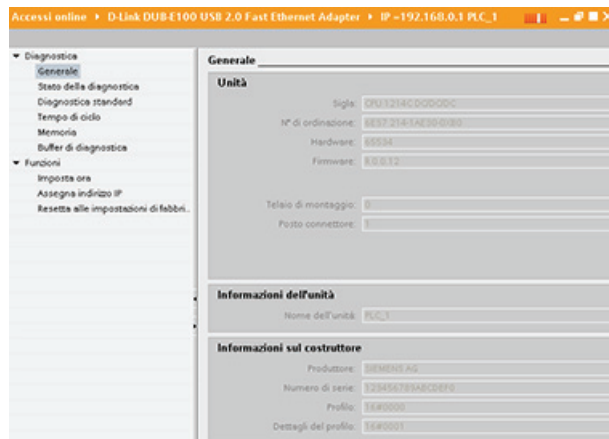
2. Selezionare le seguenti voci di menu nell'albero del progetto:

- "Accesso online"
- <Scheda adattatore per la rete nella quale si trova il dispositivo>
- "Aggiorna nodi accessibili"
- <Indirizzo dispositivo>
- "Online e diagnostica"

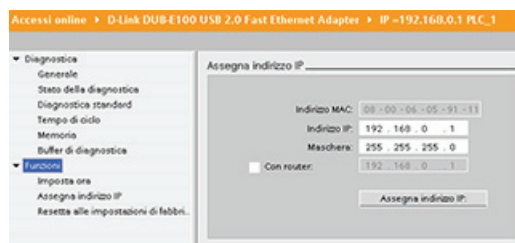


3. Selezionare le seguenti voci di menu nella finestra di dialogo "Online e diagnostica":

- "Funzioni"
- "Assegna indirizzo IP"

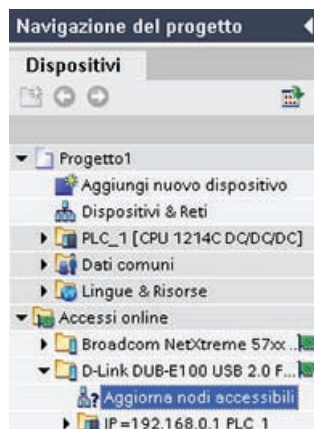


4. Specificare il nuovo indirizzo IP nel campo "Indirizzo IP".



5. Nell'albero del progetto, verificare che il nuovo indirizzo IP sia stato assegnato alla CPU selezionando le seguenti voci di menu:

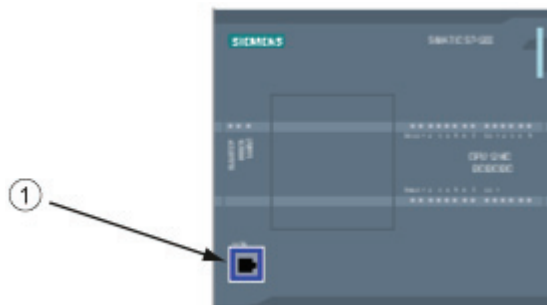
- "Accesso online"
- <Adattatore per la rete nella quale si trova il dispositivo>
- "Aggiorna nodi accessibili"



7.1.3.3 Configurazione di un indirizzo IP del progetto

Configurazione dell'interfaccia PROFINET

Dopo aver configurato il telaio di montaggio con la CPU (Pagina 226) si possono configurare i parametri per l'interfaccia PROFINET. Fare clic sul quadrato verde PROFINET della CPU e selezionare la porta PROFINET. La scheda "Proprietà" della finestra di ispezione visualizza la porta PROFINET.



① Porta PROFINET

Configurazione dell'indirizzo IP

Indirizzo Ethernet (MAC): in una rete PROFINET ciascun dispositivo è identificato da un indirizzo MAC (acronimo di Media Access Control, ovvero controllo dell'accesso al mezzo fisico) assegnato dal costruttore. Un indirizzo MAC è costituito da sei coppie di cifre esadecimali separate da tratti di congiunzione (-) o da due punti (:) disposte nell'ordine di trasmissione (ad es. 01-23-45-67-89-AB or 01:23:45:67:89:AB).

Indirizzo IP: ogni dispositivo deve avere anche un indirizzo IP (Internet Protocol) che gli consenta di fornire i dati in reti più complesse e provviste di router.

Gli indirizzi IP sono suddivisi in segmenti di 8 bit ed espressi in formato decimale separato da punti (ad esempio: 211.154.184.16). La prima parte dell'indirizzo IP corrisponde all'ID della rete (in quale rete ci si trova?) e la seconda all'ID dell'host (che è unico per ciascun dispositivo della rete). Gli indirizzi IP di tipo 192.168.x.y sono per convenzione indirizzi di reti private che non fanno parte di Internet.

Maschera della sottorete: una sottorete è un raggruppamento logico dei dispositivi collegati ad una rete. Nelle LAN (Local Area Network) i nodi delle sottoreti tendono a essere fisicamente vicini. Le maschere (chiamate "maschere di sottorete" o "maschere di rete") definiscono i limiti delle sottoreti IP.

La maschera di sottorete 255.255.255.0 è generalmente adatta a una rete locale poco estesa. Ciò significa che tutti gli indirizzi IP della rete devono avere gli stessi 3 ottetti iniziali e che i singoli dispositivi sono identificati dall'ultimo ottetto (campo di 8 bit). Ad esempio si possono contrassegnare i dispositivi di una rete locale con la maschera di sottorete 255.255.255.0 e un indirizzo IP da 192.168.2.0 a 192.168.2.255.

Le diverse sottoreti sono collegate solo tramite router. Se si utilizzano le sottoreti è quindi necessario impiegare un router IP.

Router IP: i router costituiscono il collegamento tra le LAN. Tramite il router, il computer di una LAN può trasmettere messaggi a qualsiasi altra rete che può essere a sua volta collegata ad altre LAN. Se la destinazione dei dati non si trova all'interno della LAN, il router li inoltra a un'altra rete o a gruppi di reti da dove possono essere trasmessi alla destinazione.

I router utilizzano gli indirizzi IP per trasmettere e ricevere pacchetti di dati.



Proprietà degli indirizzi IP: nella finestra Proprietà, selezionare il comando di configurazione "Indirizzo Ethernet". Il portale TIA visualizza la finestra di dialogo per la configurazione dell'indirizzo Ethernet, che consente di associare il progetto software all'indirizzo IP della CPU in cui verrà caricato il progetto.

Nota

Poiché la CPU non dispone di un indirizzo IP preconfigurato, lo si deve impostare manualmente. Se la CPU è collegata al router di una rete si deve specificare anche l'indirizzo IP del router. Durante la fase di caricamento vengono configurati tutti gli indirizzi IP.

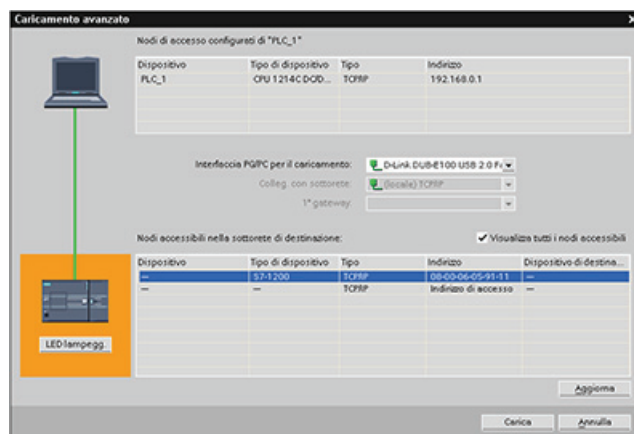
Per maggiori informazioni consultare il capitolo "Assegnazione degli indirizzi IP ai dispositivi di programmazione e ai dispositivi di rete".

La seguente tabella definisce i parametri per l'indirizzo IP:

Parametro		Descrizione
Sottorete	Nome della sottorete a cui è collegato il dispositivo. Per creare una nuova sottorete fare clic sul pulsante "Inserisci nuova sottorete". Per default è impostato "Non collegato in rete". Sono possibili due tipi di collegamento:	
	- L'opzione "Non collegato in rete" impostata per default consente di realizzare un collegamento locale. - La sottorete è necessaria se la rete contiene almeno due dispositivi.	
Protocollo IP	Indirizzo IP	Indirizzo IP assegnato alla CPU
	Maschera della sottorete	Maschera di sottorete assegnata
	Impiega router IP	Attivare la casella di controllo per indicare che si vuole utilizzare un router IP
	Indirizzo del router	Indirizzo IP assegnato al router (se applicabile)

7.1.4 Test della rete PROFINET

Una volta terminata la configurazione caricare il progetto nella CPU. Durante la fase di caricamento vengono configurati tutti gli indirizzi IP.



Assegnazione di un indirizzo IP a un dispositivo online

Poiché la CPU S7-1200 non dispone di un indirizzo IP preconfigurato, lo si deve impostare manualmente.

Per informazioni su come assegnare un indirizzo IP a un dispositivo online consultare il paragrafo "Assegnazione di un indirizzo IP online".

Per assegnare un indirizzo IP nel progetto lo si deve definire in Configurazione dispositivi e quindi salvare la configurazione e caricarla nel PLC. Per maggiori informazioni consultare "Configurazione di un indirizzo IP del progetto".

Nota

Se sono stati assegnati indirizzi IP online li si può modificare mediante la configurazione hardware online o offline.

Se gli indirizzi IP sono stati assegnati nella configurazione hardware offline, per modificare quelli assegnati dal progetto si deve utilizzare la configurazione hardware offline.

Utilizzare "Accesso online" per visualizzare l'indirizzo IP della CPU collegata come illustrato di seguito.



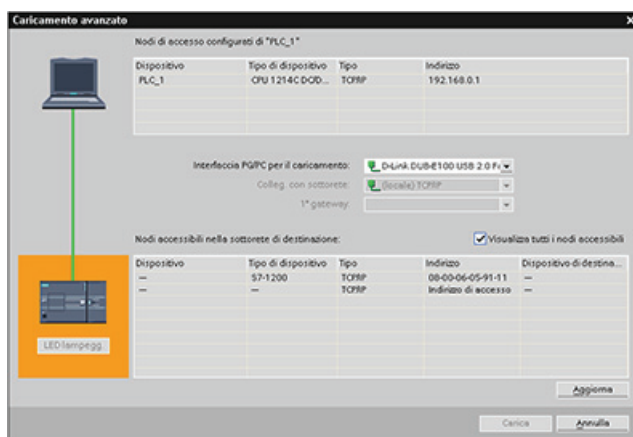
- ① La seconda delle due reti Ethernet di questo dispositivo di programmazione
- ② Indirizzo IP dell'unica CPU S7-1200 di questa rete Ethernet

Nota

Vengono visualizzate tutte le reti configurate per il dispositivo di programmazione. Per visualizzare l'indirizzo IP della CPU S7-1200 è necessario selezionare la rete corrispondente.

Utilizzo della finestra "Caricamento avanzato" per verificare i dispositivi di rete collegati

Utilizzando la funzione "Caricamento nel dispositivo" della CPU S7-1200 e la relativa finestra "Caricamento avanzato" è possibile visualizzare tutti i dispositivi di rete accessibili e verificare se vi sono stati assegnati o meno indirizzi IP univoci. Per visualizzare tutti i dispositivi accessibili e disponibili con i relativi indirizzi MAC e IP selezionare la casella "Visualizza tutti i nodi accessibili".



Se la rete desiderata non compare nell'elenco significa che per qualche motivo la comunicazione con il dispositivo in questione si è interrotta. È quindi necessario esaminare il dispositivo e la rete per individuare eventuali errori hardware e/o di configurazione.

7.2 Comunicazione da HMI a PLC



La CPU supporta i collegamenti con gli HMI per la comunicazione PROFINET. In fase di configurazione della comunicazione tra CPU e HMI è necessario tener conto dei seguenti requisiti:

Configurazione/impostazione:

- La porta PROFINET della CPU deve essere configurata in modo da collegarsi all'HMI.
- L'HMI deve essere impostato e configurato.
- Le informazioni di configurazione dell'HMI fanno parte del progetto della CPU e possono essere configurate e scaricate dal progetto stesso.
- Per la comunicazione diretta tra due dispositivi non è necessario utilizzare uno switch Ethernet, che è invece indispensabile se si collegano in rete più di due dispositivi.

Nota

Per collegare le CPU e i dispositivi HMI è possibile utilizzare lo switch Ethernet a 4 porte CSM1277 Siemens montato su telaio. La porta PROFINET della CPU non contiene un dispositivo di commutazione Ethernet.

Funzioni supportate:

- L'HMI è in grado di accedere ai dati della CPU sia in lettura che in scrittura.
- È possibile attivare dei messaggi sulla base delle informazioni recuperate dalla CPU.
- Diagnostica di sistema

Nota

WinCC Basic e STEP 7 Basic sono componenti del portale TIA. Per ulteriori informazioni sulla configurazione dell'HMI fare riferimento a WinCC Basic.

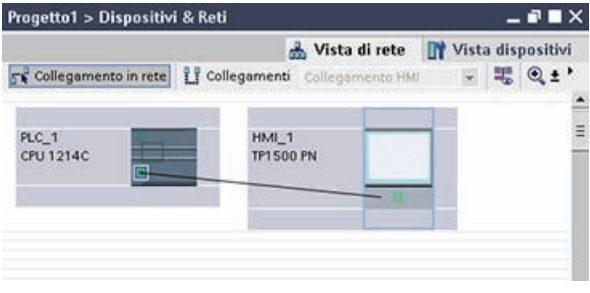
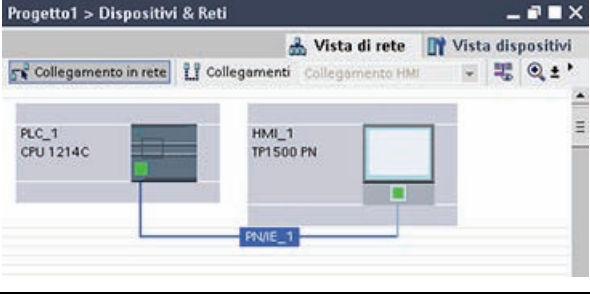
Procedura di configurazione della comunicazione tra un HMI e una CPU

Operazione	Task
1	Realizzazione del collegamento hardware per la comunicazione Il collegamento fisico tra un HMI e una CPU è costituito da un'interfaccia PROFINET. Poiché la CPU dispone della funzionalità Autocrossover, per l'interfaccia è possibile utilizzare indifferentemente un cavo Ethernet di tipo standard o crossover. Questo tipo di collegamento non richiede uno switch Ethernet. Per maggiori informazioni consultare il paragrafo "Comunicazione con un dispositivo di programmazione: Realizzazione del collegamento hardware per la comunicazione" (Pagina 225).
2	Configurazione dei dispositivi Per maggiori informazioni consultare il paragrafo "Comunicazione con un dispositivo di programmazione: Configurazione dei dispositivi" (Pagina 226).
3	Configurazione dei collegamenti logici di rete tra un HMI e una CPU Per maggiori informazioni consultare il paragrafo "Comunicazione da HMI a PLC: Configurazione dei collegamenti logici di rete tra un HMI e una CPU" (Pagina 236).
4	Configurazione di un indirizzo IP del progetto Utilizzare lo stesso procedimento di configurazione, ricordando tuttavia che è necessario configurare gli indirizzi IP per l'HMI e la CPU. Per maggiori informazioni consultare il paragrafo "Comunicazione con un dispositivo di programmazione: Configurazione di un indirizzo IP del progetto" (Pagina 231).
5	Test della rete PROFINET È necessario caricare la configurazione per ciascuna CPU. Per maggiori informazioni consultare il paragrafo "Comunicazione con un dispositivo di programmazione: Test della rete PROFINET" (Pagina 233).

7.2.1 Configurazione dei collegamenti logici di rete tra un HMI e una CPU

Dopo avere configurato il telaio di montaggio con la CPU è possibile procedere con la configurazione dei collegamenti di rete.

Nel portale Dispositivi e reti, utilizzare "Vista di rete" per creare i collegamenti di rete tra i dispositivi inseriti nel progetto. Per creare un collegamento Ethernet selezionare la casella verde (Ethernet) sulla CPU e trascinarla verso il dispositivo HMI tracciando una linea di congiunzione. Quindi rilasciare il pulsante del mouse.

Azione	Risultato
Selezionare "Vista di rete" per visualizzare i dispositivi da collegare.	
Selezionare la porta di un dispositivo e trascinare il collegamento nella porta del secondo dispositivo.	
Rilasciare il tasto del mouse per creare il collegamento.	

7.3 Comunicazione da PLC a PLC



Una CPU può comunicare con un'altra CPU di una rete mediante le istruzioni TSEND_C e TRCV_C.

Quando si configura la comunicazione tra due CPU è necessario tener conto di quanto segue:

- Configurazione/impostazione: è necessario effettuare la configurazione hardware.
- Funzioni supportate: accesso in lettura/scrittura ai dati di una CPU paritaria
- Per la comunicazione diretta tra due dispositivi non è necessario utilizzare uno switch Ethernet, che è invece indispensabile se si collegano in rete più di due dispositivi.

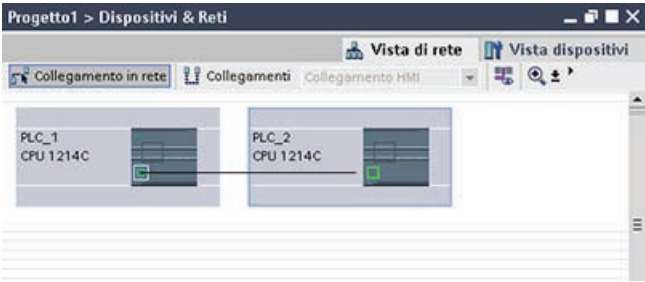
Procedura di configurazione della comunicazione tra due CPU

Operazione	Task
1	Realizzazione del collegamento hardware per la comunicazione Il collegamento fisico tra due CPU è costituito da un'interfaccia PROFINET. Poiché la CPU dispone della funzionalità Autocrossover, per l'interfaccia è possibile utilizzare indifferentemente un cavo Ethernet di tipo standard o crossover. Questo tipo di collegamento non richiede uno switch Ethernet. Per maggiori informazioni consultare il paragrafo "Comunicazione con un dispositivo di programmazione: Realizzazione del collegamento hardware per la comunicazione".
2	Configurazione dei dispositivi È necessario configurare due progetti che contengono ognuno una CPU. Per maggiori informazioni consultare il paragrafo "Comunicazione con un dispositivo di programmazione: Configurazione dei dispositivi".
3	Configurazione dei collegamenti logici di rete tra due CPU Per ulteriori informazioni consultare il paragrafo "Configurazione della comunicazione tra due CPU: Configurazione dei collegamenti logici di rete tra due CPU" (Pagina 238).
4	Configurazione di un indirizzo IP del progetto Utilizzare lo stesso procedimento di configurazione, ricordando tuttavia che è necessario configurare gli indirizzi IP per due CPU (ad esempio PLC_1 e PLC_2). Per maggiori informazioni consultare il paragrafo "Comunicazione con un dispositivo di programmazione: Configurazione di un indirizzo IP del progetto".
5	Configurazione dei parametri di trasmissione (invio) e di ricezione Perché due CPU possano comunicare tra loro è necessario configurare le istruzioni TSEND_C e TRCV_C su entrambe. Per ulteriori informazioni consultare il paragrafo "Configurazione della comunicazione tra due CPU: Configurazione dei parametri di trasmissione (invio) e di ricezione" (Pagina 239).
6	Test della rete PROFINET È necessario caricare la configurazione per ciascuna CPU. Per ulteriori informazioni consultare il paragrafo "Configurazione della comunicazione tra un dispositivo di programmazione e una CPU: Test della rete PROFINET".

7.3.1 Configurazione dei collegamenti logici di rete tra due CPU

Dopo avere configurato il telaio di montaggio con la CPU è possibile procedere con la configurazione dei collegamenti di rete.

Nel portale Dispositivi e reti, utilizzare "Vista di rete" per creare i collegamenti di rete tra i dispositivi inseriti nel progetto. Per creare un collegamento PROFINET selezionare la casella verde (PROFINET) sulla prima CPU e trascinarla verso il secondo PLC tracciando una linea di congiunzione. Quindi rilasciare il pulsante del mouse.

Azione	Risultato
Selezionare "Vista di rete" per visualizzare i dispositivi da collegare.	
Selezionare la porta di un dispositivo e trascinare il collegamento nella porta del secondo dispositivo.	
Rilasciare il tasto del mouse per creare il collegamento.	

7.3.2 Configurazione dei parametri di trasmissione (invio) e di ricezione

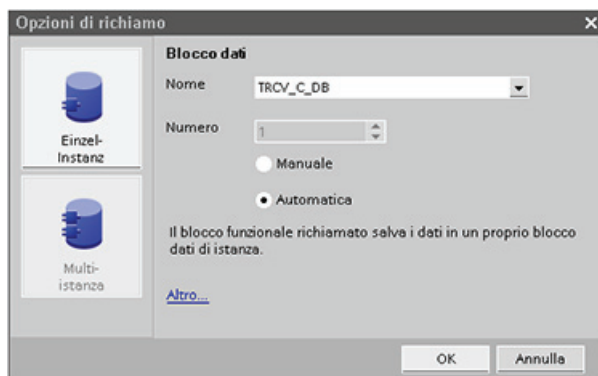
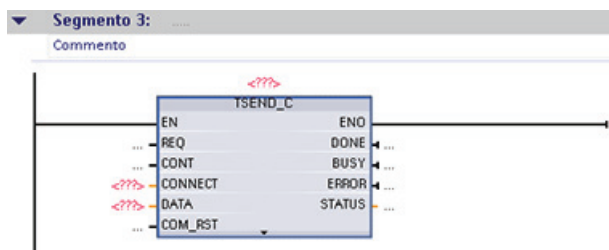
Per creare i collegamenti tra due CPU si utilizza la comunicazione tramite blocco di trasmissione (blocco T). Perché le CPU possano stabilire una comunicazione PROFINET è necessario configurare i parametri per la trasmissione (o invio) e la ricezione dei messaggi. Questi parametri determinano le modalità di comunicazione per la ricezione e la trasmissione dei messaggi da e verso un dispositivo di destinazione.

7.3.2.1 Configurazione dei parametri di trasmissione (invio) dell'istruzione TSEND_C

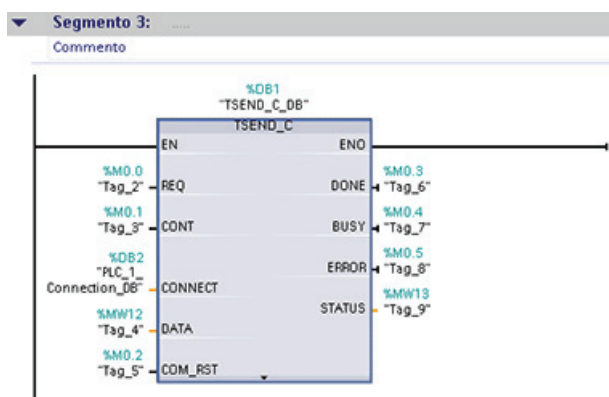
Istruzione TSEND_C

L'istruzione TSEND_C (Pagina 164) crea un collegamento per la comunicazione con una stazione partner. Il collegamento viene configurato, creato e controllato automaticamente finché l'istruzione non ne determina l'interruzione. L'istruzione TSEND_C riunisce in sé le funzioni delle istruzioni TCON, TDISCON e TSEND.

From the Device configurazione in STEP 7 Basic, you can configure how a TSEND_C instruction transmits data. Innanzitutto è necessario inserire l'istruzione nel programma dalla cartella "Comunicazione" nelle "Istruzioni avanzate". TSEND_C viene visualizzata insieme alla finestra Opzioni di richiamo che consente di assegnare un DB per la memorizzazione dei parametri dell'istruzione.



È possibile assegnare agli ingressi e alle uscite indirizzi della memoria delle variabili, come illustrato nella figura che segue.



Configurazione dei parametri generali

I parametri di comunicazione devono essere specificati nella finestra di configurazione delle proprietà dell'istruzione TSEND_C. La finestra compare vicino al bordo inferiore della pagina ogni volta che si seleziona una parte qualsiasi dell'istruzione TSEND_C.

Configurazione dei parametri di collegamento

Ogni CPU ha una porta PROFINET integrata che supporta la comunicazione PROFINET standard. I protocolli Ethernet supportati sono descritti nei due tipi di collegamento seguenti:

Protocollo	Nome del protocollo	Utilizzo
RFC 1006	ISO on TCP	Frammentazione e ricomposizione dei messaggi
TCP	Transport Control Protocol	Trasporto di frame

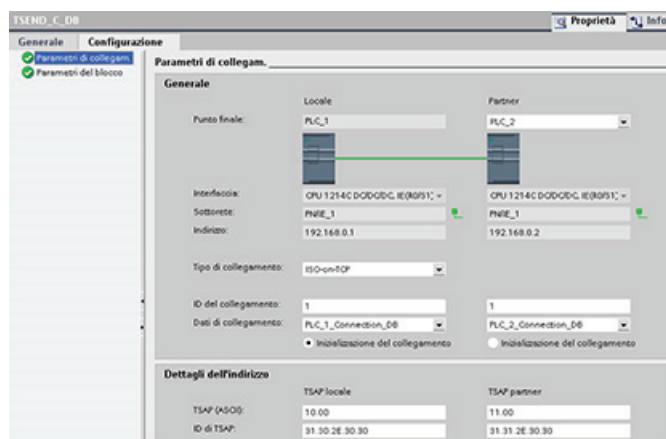
ISO on TCP (RFC 1006)

ISO on TCP è un meccanismo che permette di trasferire le applicazioni ISO sulla rete TCP/IP. Il protocollo ha le seguenti caratteristiche:

- Efficiente protocollo di comunicazione strettamente collegato all'hardware
- Idoneo per il trasferimento di quantità di dati medio-grandi (fino a 8192 byte)
- Contrariamente a quanto accade con il TCP, i messaggi sono caratterizzati da un identificativo di fine dati e sono orientati al messaggio.
- Supporta la funzione di routing; è utilizzabile nelle WAN
- Sono ammesse lunghezze di dati dinamiche.
- L'interfaccia di programmazione SEND / RECEIVE richiede determinate operazioni di programmazione per la gestione dei dati.

Grazie all'utilizzo dei punti di accesso TSAP (Transport Service Access Point), il protocollo TCP consente collegamenti multipli a un unico indirizzo IP (collegamenti fino a 64 K). Con RFC 1006, i TSAP identificano in modo univoco i collegamenti di questi punti finali di comunicazione ad un indirizzo IP.

Nella sezione "Dettagli dell'indirizzo" della finestra Parametri di collegamento è possibile definire i TSAP da utilizzare. Nel campo "TSAP locale" si inserisce il TSAP di un collegamento nella CPU. Il TSAP assegnato per il collegamento nella CPU partner viene invece inserito nel campo "TSAP del partner".



Parametro	Definizione
Dati generali	
Punto finale: Partner	Nome assegnato alla CPU partner (ricevente)
Interfaccia	Nome assegnato alle interfacce
Sottorete	Nome assegnato alle sottoreti
Indirizzo	Indirizzi IP assegnati
Tipo di collegamento	Tipo di protocollo Ethernet
ID del collegamento	Numero di ID
Dati di collegamento	Indirizzo di memoria dei dati della CPU locale e partner
Inizializzazione del collegamento	Pulsante per la selezione della CPU locale o partner come collegamento attivo
Dettagli dell'indirizzo	
TSAP ¹ (ASCII)	TSAP della CPU locale e partner in formato ASCII
ID di TSAP	TSAP della CPU locale e partner in formato esadecimale

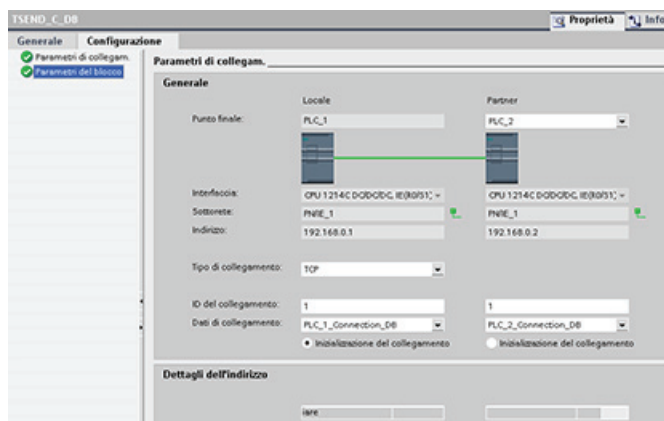
¹ Quando si configura un collegamento con una CPU S7-1200 tramite ISO on TCP, utilizzare solo caratteri ASCII nell'estensione TSAP per i partner di comunicazione passivi.

Protocollo di controllo TCP (Transport Control Protocol)

Il TCP è un protocollo standard descritto dall'RFC 793: Transmission Control Protocol. Lo scopo principale del TCP è fornire un servizio di collegamento sicuro e affidabile tra coppie di processi. Il protocollo ha le seguenti caratteristiche:

- Efficiente protocollo di comunicazione strettamente collegato all'hardware
- Idoneo per il trasferimento di quantità di dati medio-grandi (fino a 8192 byte)
- Assicura maggiore efficienza a livello applicativo perché offre:
 - Recupero degli errori
 - Controllo del flusso
 - Affidabilità
- Protocollo orientato al collegamento
- Garantisce la massima flessibilità di utilizzo con sistemi di altri produttori che supportano esclusivamente il TCP
- Supporta la funzione di routing

- Sono ammesse solo lunghezze di dati statiche
- Conferma della ricezione dei messaggi
- Indirizzamento delle applicazioni tramite i numeri delle porte.
- La maggior parte dei protocolli delle applicazioni utente, quali TELNET e FTP, utilizzano TCP.
- L'interfaccia di programmazione SEND / RECEIVE richiede determinate operazioni di programmazione per la gestione dei dati.



Parametro	Definizione
Dati generali	
Punto finale: Partner	Nome assegnato alla CPU partner (ricevente)
Interfaccia	Nome assegnato alle interfacce
Sottorete	Nome assegnato alle sottoreti
Indirizzo	Indirizzi IP assegnati
Tipo di collegamento	Tipo di protocollo Ethernet
ID del collegamento	Numero di ID
Dati di collegamento	Indirizzo di memoria dei dati della CPU locale e partner
Inizializzazione del collegamento	Pulsante per la selezione della CPU locale o partner come collegamento attivo
Dettagli dell'indirizzo	
Porta (decimale)	Porta della CPU partner in formato decimale

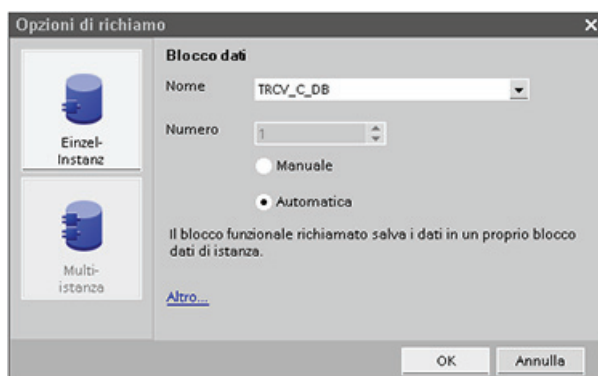
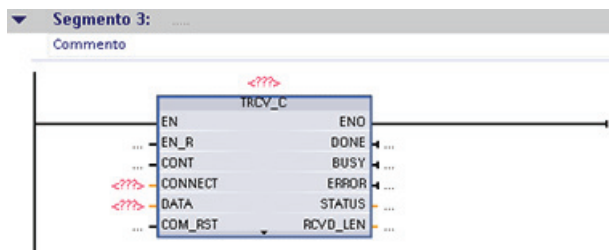
7.3.2.2 Configurazione dei parametri di ricezione dell'istruzione TRCV_C

Istruzione TRCV_C

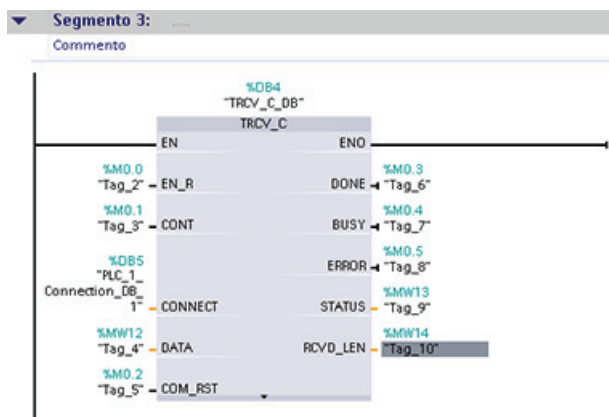
L'istruzione TRCV_C (Pagina 164) crea un collegamento per la comunicazione con una stazione partner. Il collegamento viene configurato, creato e controllato automaticamente finché l'istruzione non ne determina l'interruzione. L'istruzione TRCV_C riunisce le funzioni delle istruzioni TCON, TDISCON e TRCV.

Nella finestra Configurazione dispositivi di STEP 7 Basic si può configurare in che modo l'istruzione TSEND_C trasmette i dati. Innanzitutto è necessario inserire l'istruzione nel

programma dalla cartella "Comunicazione" nelle "Istruzioni avanzate". Insieme all'istruzione TRCV_C viene visualizzata la finestra Opzioni di richiamo, nella quale si assegna un DB per il salvataggio dei parametri corrispondenti.



È possibile assegnare agli ingressi e alle uscite indirizzi della memoria delle variabili, come illustrato nella figura che segue.



Configurazione dei parametri generali

I parametri di comunicazione devono essere specificati nella finestra di configurazione delle proprietà dell'istruzione TRCV_C. Questa finestra compare vicino al bordo inferiore della pagina ogni volta che si seleziona una parte qualsiasi dell'istruzione TRCV_C.

Configurazione dei parametri di collegamento

Ogni CPU ha una porta PROFINET integrata che supporta la comunicazione PROFINET standard. I protocolli Ethernet supportati sono descritti nei due tipi di collegamento seguenti:

Protocollo	Nome del protocollo	Utilizzo
RFC 1006	ISO on TCP	Frammentazione e ricomposizione dei messaggi
TCP	Transport Control Protocol	Trasporto di frame

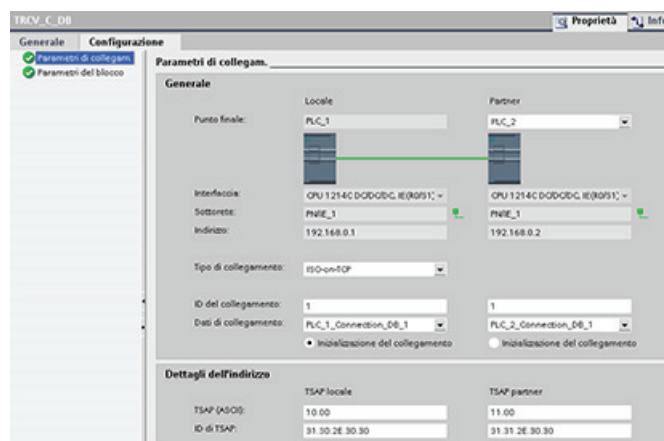
ISO on TCP (RFC 1006)

ISO on TCP è un meccanismo che permette di trasferire le applicazioni ISO sulla rete TCP/IP. Il protocollo ha le seguenti caratteristiche:

- Efficiente protocollo di comunicazione strettamente collegato all'hardware
- Idoneo per il trasferimento di quantità di dati medio-grandi (fino a 8192 byte)
- Contrariamente a quanto accade con il TCP, i messaggi sono caratterizzati da un identificativo di fine dati e sono orientati al messaggio.
- Supporta la funzione di routing; è utilizzabile nelle WAN
- Sono ammesse lunghezze di dati dinamiche.
- L'interfaccia di programmazione SEND / RECEIVE richiede determinate operazioni di programmazione per la gestione dei dati.

Grazie all'utilizzo dei punti di accesso TSAP (Transport Service Access Point), il protocollo TCP consente collegamenti multipli a un unico indirizzo IP (collegamenti fino a 64 K). Con RFC 1006, i TSAP identificano in modo univoco i collegamenti di questi punti finali di comunicazione ad un indirizzo IP.

Nella sezione "Dettagli dell'indirizzo" della finestra Parametri di collegamento è possibile definire i TSAP da utilizzare. Nel campo "TSAP locale" si inserisce il TSAP di un collegamento nella CPU. Il TSAP assegnato per il collegamento nella CPU partner viene invece inserito nel campo "TSAP del partner".



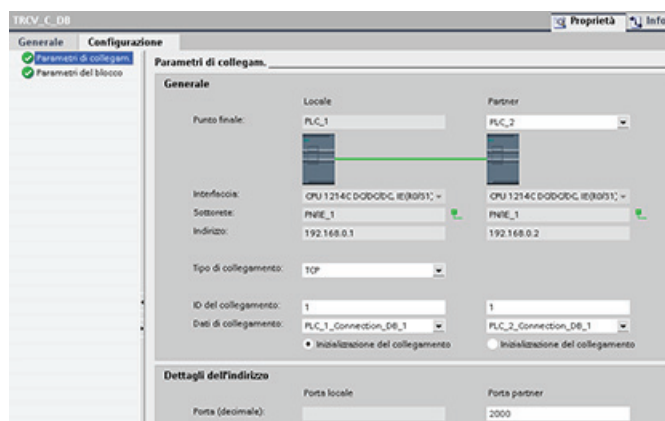
Parametro	Definizione
Dati generali	
Punto finale: Partner	Nome assegnato alla CPU partner (ricevente)
Interfaccia	Nome assegnato alle interfacce
Sottorete	Nome assegnato alle sottoreti
Indirizzo	Indirizzi IP assegnati
Tipo di collegamento	Tipo di protocollo Ethernet
ID del collegamento	Numero di ID
Dati di collegamento	Indirizzo di memoria dei dati della CPU locale e partner
Inizializzazione del collegamento	Pulsante per la selezione della CPU locale o partner come collegamento attivo
Dettagli dell'indirizzo	
TSAP ¹ (ASCII)	TSAP della CPU locale e partner in formato ASCII
ID di TSAP	TSAP della CPU locale e partner in formato esadecimale

¹ Quando si configura un collegamento con una CPU S7-1200 tramite ISO on TCP, utilizzare solo caratteri ASCII nell'estensione TSAP per i partner di comunicazione passivi.

Protocollo di controllo TCP (Transport Control Protocol)

Il TCP è un protocollo standard descritto dall'RFC 793: Transmission Control Protocol. Lo scopo principale del TCP è fornire un servizio di collegamento sicuro e affidabile tra coppie di processi. Il protocollo ha le seguenti caratteristiche:

- Efficiente protocollo di comunicazione strettamente collegato all'hardware
- Idoneo per il trasferimento di quantità di dati medio-grandi (fino a 8192 byte)
- Assicura maggiore efficienza a livello applicativo perché offre:
 - Recupero degli errori
 - Controllo del flusso
 - Affidabilità
- Protocollo orientato al collegamento
- Garantisce la massima flessibilità di utilizzo con sistemi di altri produttori che supportano esclusivamente il TCP
- Supporta la funzione di routing
- Sono ammesse solo lunghezze di dati statiche
- Conferma della ricezione dei messaggi
- Indirizzamento delle applicazioni tramite i numeri delle porte
- La maggior parte dei protocolli delle applicazioni utente, quali TELNET e FTP, utilizzano TCP.
- L'interfaccia di programmazione SEND / RECEIVE richiede determinate operazioni di programmazione per la gestione dei dati.



Parametro	Definizione
Dati generali	
Punto finale: Partner	Nome assegnato alla CPU partner (ricevente)
Interfaccia	Nome assegnato alle interfacce
Sottorete	Nome assegnato alle sottoreti
Indirizzo	Indirizzi IP assegnati
Tipo di collegamento	Tipo di protocollo Ethernet
ID del collegamento	Numero di ID
Dati di collegamento	Indirizzo di memoria dei dati della CPU locale e partner
Inizializzazione del collegamento	Pulsante per la selezione della CPU locale o partner come collegamento attivo
Dettagli dell'indirizzo	
Porta (decimale)	Porta della CPU locale in formato decimale

7.4 Informazioni di riferimento

7.4.1 Posizione dell'indirizzo Ethernet (MAC) sulla CPU

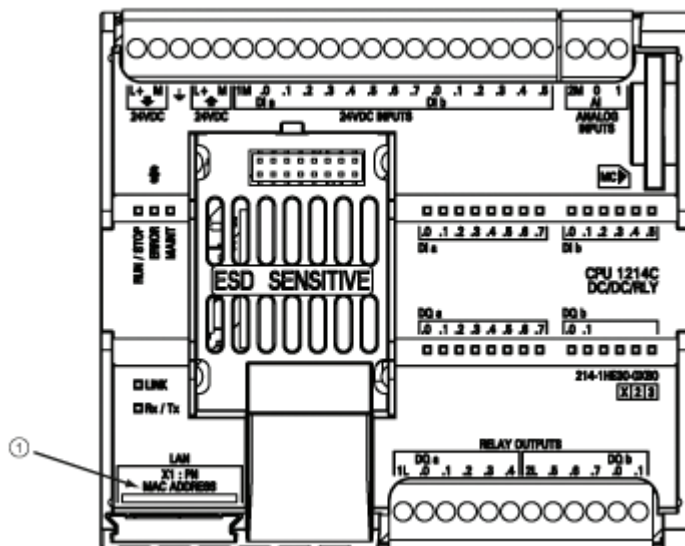
Nell'ambito delle reti PROFINET, un indirizzo MAC (Media Access Control) è un identificatore usato dal produttore per identificare le schede adattatori. Normalmente un indirizzo MAC codifica il numero identificativo registrato del produttore.

Il formato standard (IEEE 802.3) per la stampa degli indirizzi MAC in formato comprensibile all'uomo è costituito da sei gruppi di cifre esadecimali separate da trattini di congiunzione (-) o due punti (:) e disposti nell'ordine di trasmissione (ad esempio, 01-23-45-67-89-ab o 01:23:45:67:89:ab).

Nota

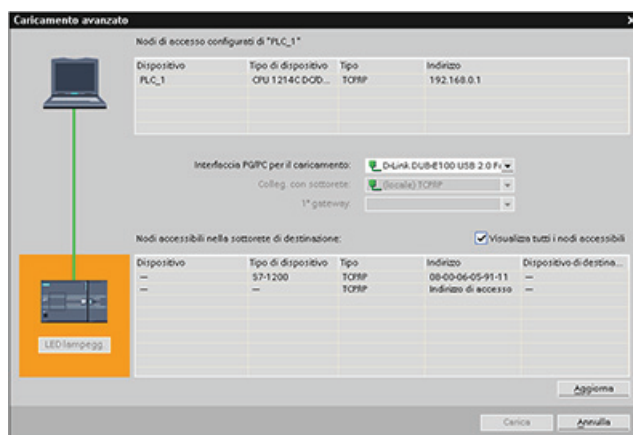
Ogni CPU viene fornita con un indirizzo MAC univoco e permanente predefinito che non può essere modificato dall'utente.

L'indirizzo MAC è impresso in basso a sinistra sul lato anteriore della CPU. Si noti che per vedere le informazioni sull'indirizzo MAC è necessario sollevare le coperture TB inferiori.



① Indirizzo MAC

Inizialmente la CPU non dispone di un indirizzo IP, ma solo dell'indirizzo MAC predefinito dal produttore. Per la comunicazione PROFINET è indispensabile che ai dispositivi sia assegnato un indirizzo IP univoco.



Utilizzando la funzione "Caricamento nel dispositivo" della CPU e la finestra "Caricamento avanzato" è possibile visualizzare tutti i dispositivi di rete accessibili e assicurarsi che ad ognuno di questi sia stato assegnato un indirizzo IP univoco. La finestra visualizza tutti i dispositivi accessibili e disponibili con i corrispondenti indirizzi MAC e IP. Gli indirizzi MAC sono fondamentali per identificare i dispositivi ai quali non è stato assegnato il necessario indirizzo IP univoco.

7.4.2 Configurazione della sincronizzazione del Network Time Protocol

Il Network Time Protocol (NTP) è un protocollo molto diffuso per sincronizzare gli orologi dei computer in base ai time server di Internet. Offre un'accuratezza di sincronizzazione con margini tipicamente inferiori al millisecondo nelle reti LAN e di pochi millisecondi nelle reti

WAN. Le configurazioni NTP tipiche usano server a ridondanza multipla e diversi percorsi di rete per offrire una precisione e un'affidabilità elevate.

La sottorete NTP opera con una gerarchia a più livelli dove ad ogni livello è assegnato un numero detto "strato". I server di strato 1 (primario), al primo livello, sono sincronizzati direttamente con i servizi orari nazionali. I server di strato 2 (secondario), di livello successivo, sono sincronizzati con i server dello strato 1 e così via.

Parametri per la sincronizzazione dell'ora

Nella finestra Proprietà, selezionare il comando di configurazione "Sincronizzazione dell'ora". Il portale TIA visualizza la finestra di configurazione Sincronizzazione dell'ora:

Sincronizzazione dell'ora

☒ Sincronizzazione dell'ora nella procedura NTP

Server 1: 192 . 168 . 0 .21

Server 2: 192 . 168 . 0 .22

Server 3: 192 . 168 . 0 .23

Server 4: 192 . 168 . 0 .24

Intervallo di aggiornamento: 10 sec

Nota

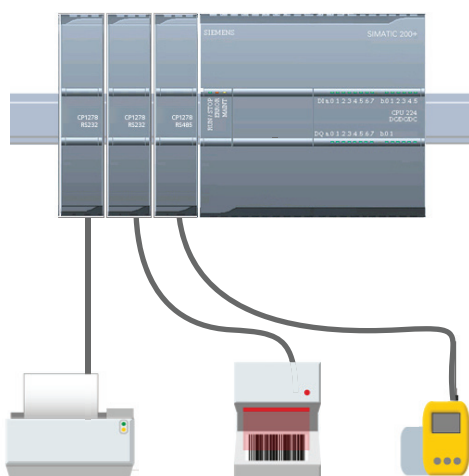
Durante la fase di caricamento vengono configurati tutti gli indirizzi IP.

La seguente tabella definisce i parametri per la sincronizzazione dell'ora:

Parametro	Definizione
Consente la sincronizzazione dell'ora mediante i server Network Time Protocol (NTP)	Attivare la casella di controllo per consentire la sincronizzazione dell'ora tramite i server NTP.
Server 1	Indirizzo IP assegnato al network time server 1
Server 2	Indirizzo IP assegnato al network time server 2
Server 3	Indirizzo IP assegnato al network time server 3
Server 4	Indirizzo IP assegnato al network time server 4
Intervallo per la sincronizzazione dell'ora	Valore dell'intervallo (sec)

Comunicazioni punto a punto (PtP)

La CPU supporta il protocollo punto a punto (PtP) per la comunicazione seriale basata su caratteri in cui l'applicazione utente definisce e implementa completamente il protocollo scelto. La PtP offre la massima libertà e flessibilità, ma richiede una complessa implementazione nel programma utente.



Il PtP offre un'ampia gamma di possibilità:

- La possibilità di inviare informazioni direttamente ad un dispositivo esterno, ad es. una stampante
- La possibilità di ricevere informazioni da altri dispositivi, come lettori di codici a barre, lettori RFID, macchine fotografiche o sistemi di visione di terzi e molti altri tipi di dispositivi
- La possibilità di scambiare informazioni, inviare e ricevere dati da altri dispositivi come GPS, macchine fotografiche o sistemi di visione di terzi, modem radio e molti altri

La PtP è una comunicazione seriale che utilizza UART standard a supporto di una vasta gamma di velocità di trasmissione e di opzioni di parità. Il modulo di comunicazione (CM) RS232 o RS485 rappresenta l'interfaccia elettrica per le comunicazioni PtP.

STEP 7 Basic offre biblioteche di istruzioni che l'utente può utilizzare per programmare la propria applicazione. Queste biblioteche dispongono di funzioni di comunicazione PtP per i seguenti protocolli:

- Protocollo per azionamenti USS
- Protocollo master Modbus RTU
- Protocollo slave Modbus RTU

8.1 Uso dei moduli di comunicazione RS232 e RS485

Due moduli di comunicazione (CM) costituiscono l'interfaccia per le comunicazioni PtP: CM 1241 RS485 (Pagina 335) e CM 1241 RS232 (Pagina 336). È possibile collegare fino a tre CM (di qualsiasi tipo). Installare il CM a sinistra della CPU o di un altro CM. Per maggiori informazioni sull'installazione o lo smontaggio dei moduli consultare il paragrafo "Montaggio" (Pagina 33).

I moduli di comunicazione RS232 e RS485 hanno le seguenti caratteristiche:

- Porta isolata
- Supporto dei protocolli PtP

- Configurazione e programmazione tramite istruzioni avanzate e funzioni della biblioteca
- LED per la visualizzazione dell'attività di trasmissione e ricezione
- LED per la diagnostica
- Alimentazione tramite la CPU, non è necessario un collegamento esterno per l'alimentazione

Consultare il capitolo relativo ai dati tecnici dei moduli di comunicazione (Pagina 335).

8.2 Configurazione delle porte di comunicazione

I moduli di comunicazione possono essere configurati nei due seguenti modi:

- Nella finestra Configurazione dispositivo di STEP 7 Basic che consente di configurare i parametri delle porte (velocità di trasmissione e parità), di trasmissione e di ricezione. Le impostazioni della finestra vengono salvate in modo permanente nella CPU e vengono applicate in seguito allo spegnimento/riaccensione e alla commutazione da RUN a STOP.
- Per impostare i parametri utilizzare le istruzioni PORT_CFG, SEND_CFG e RCV_CFG. Le impostazioni delle porte effettuate dalle istruzioni sono valide quando la CPU è in RUN. e vengono ripristinate su quelle di Configurazione dispositivo dopo la commutazione in STOP o lo spegnimento/riaccensione.

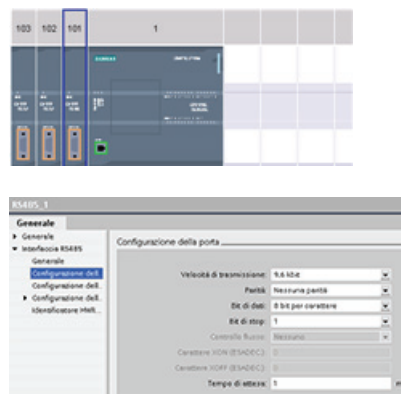
Dopo avere configurato i dispositivi hardware (Pagina 77), configurare i parametri delle interfacce di comunicazione selezionando uno dei CM dal telaio di montaggio.

La scheda "Proprietà" della finestra di ispezione visualizza i parametri del CM selezionato. Selezionare "Configurazione della porta" per modificare i seguenti parametri:

- Velocità di trasmissione
- Parità
- Numero di bit di stop
- Controllo del flusso (solo RS232)
- Tempo di attesa

Tranne che per il controllo del flusso, i parametri di configurazione della porta sono gli stessi sia per il modulo RS232 che per il modulo RS485. I valori dei parametri possono invece essere diversi.

Si può configurare la porta (o modificare la configurazione esistente) anche dal programma utente utilizzando l'istruzione PORT_CFG (Pagina 264).



Nota

I valori dei parametri impostati dall'istruzione PORT_CFG nel programma utente sovrascrivono le impostazioni della porta effettuate da STEP 7 Basic. Si segnala che l'S7-1200 non mantiene i parametri impostati con l'istruzione PORT_CFG in caso di mancanza dell'alimentazione.

Velocità di trasmissione: il valore di default della velocità di trasmissione (baud rate) è di 9,6 kbit al secondo. Sono opzioni valide:

300 baud	2,4 kbit	19,2 kbit	76,8 kbit
600 baud	4,8 kbit	28,4 kbit	115,2 kbit
1,2 kbit	9,6 kbit	57,6 kbit	

Parità: Il valore di default per la parità è nessuna parità. Sono opzioni valide:

- Nessuna parità
- Parità pari
- Parità dispari
- Parità mark (bit di parità sempre impostato a 1)
- Parità space (bit di parità sempre impostato a 0)

Numero di bit di stop: I bit di stop possono essere uno o due. Il valore di default è uno.

Controllo di flusso: Per il modulo di comunicazione RS232 è possibile selezionare un controllo di tipo hardware o software, come descritto nel paragrafo "Gestione del controllo di flusso (Pagina 253)". Se si opta per il controllo del flusso hardware è possibile scegliere tra segnale RTS sempre ON oppure disattivato. Se si opta per il controllo del flusso software è possibile definire i caratteri ASCII per i caratteri XON e XOFF.

Il modulo di comunicazione RS485 non supporta il controllo del flusso.

Tempo di attesa: A seconda del tipo di controllo del flusso, il tempo di attesa indica l'intervallo di tempo che il modulo di comunicazione attende per ricevere un CTS dopo la conferma di un RTS o per ricevere un XON dopo avere ricevuto un XOFF. Se il tempo di attesa termina prima che il modulo di comunicazione abbia ricevuto il CTS o il XON atteso, il modulo annulla l'operazione di trasmissione e segnala un errore al programma utente. Specificare il tempo di attesa in millisecondi nel campo compreso tra 0 e 65535 millisecondi.

8.3 Gestione del controllo di flusso

Il controllo del flusso è un meccanismo che serve per bilanciare le trasmissioni di invio e ricezione di dati, in modo tale che non vadano persi dati. Il controllo del flusso garantisce che un dispositivo di trasmissione non invii più informazioni di quelle che il dispositivo ricevente può gestire. Il controllo del flusso può essere realizzato via hardware o software. Il CM RS232 supporta entrambi i tipi di controllo, mentre il CM RS485 non lo supporta. Il tipo di controllo del flusso si specifica in fase di configurazione della porta (Pagina 252) o con l'istruzione PORT_CFG.

Il controllo del flusso hardware funziona per mezzo dei segnali di comunicazione Request to send (richiesta di invio) (RTS) e Clear to send (pronto a trasmettere) (CTS). Con il modulo RS232, il segnale RTS viene emesso dal pin 7 e il segnale CTS viene ricevuto dal pin 8. Il CM 1241 è un DTE (Data Terminal Equipment) che invia un RTS in uscita e controlla il CTS in ingresso.

Controllo del flusso hardware: RTS disattivato

Abilitando il controllo del flusso hardware con RTS disattivato per un CM RS232, il modulo attiva il segnale RTS per l'invio dei dati. Il modulo controlla il segnale CTS per determinare se il dispositivo ricevente può accettare i dati. Quando il segnale CTS è attivo, il modulo può

trasmettere dati fintanto che il segnale CTS resta attivo. Se il segnale CTS si disattiva la trasmissione deve interrompersi.

La trasmissione riprende quando il segnale CTS si riattiva. Se il segnale CTS non si attiva entro il tempo di attesa configurato, il modulo annulla la trasmissione e segnala un errore al programma utente. Specificare il tempo di attesa nella configurazione della porta (Pagina 252).

L'attivazione del controllo del flusso mediante RTS è utile nei dispositivi che richiedono un segnale che indichi che la trasmissione è attiva. Ad esempio un modem radio che si serve dell'RTS come segnale "chiave" per alimentare il trasmettitore. Questo tipo di controllo del flusso non funziona con i normali modem telefonici. Per questo tipo di modem si deve utilizzare l'impostazione RTS sempre on.

Controllo del flusso hardware: RTS sempre ON

Nel modo RTS sempre on il CM 1241 attiva per default il segnale RTS. Il dispositivo, ad es. un modem telefonico, controlla il segnale RTS dal CM e lo utilizza come CTS (pronto a trasmettere). Il modem trasmette al CM solo quando l'RTS è attivo, ovvero quando il modem telefonico rileva un CTS attivo. Se l'RTS è disattivato il modem telefonico non trasmette al CM.

Per consentire al modem di inviare dati al CM in qualsiasi momento, configurare il controllo del flusso hardware "RTS sempre ON". In questo modo il CM attiva sempre il segnale RTS. Il CM non disattiva RTS nemmeno quando il modulo non può accettare caratteri. Il dispositivo di trasmissione deve verificare di non causare un overrun del buffer di ricezione del CM.

Utilizzo dei segnali Data Terminal Block Ready (DTR) e Data Set Ready (DSR)

Il CM attiva DTR per uno dei due tipi di controllo del flusso hardware. Il modulo trasmette soltanto quando il segnale DSR si attiva. Lo stato del DSR viene valutato soltanto all'inizio dell'operazione di invio. Se il DSR si disattiva dopo l'inizio della trasmissione, quest'ultima non viene interrotta.

Controllo del flusso software

Il controllo del flusso software usa caratteri speciali nei messaggi per controllare il flusso, ovvero dei caratteri ASCII che rappresentano XON e XOFF.

XOFF indica che una trasmissione deve interrompersi. XON indica che una trasmissione può riprendere.

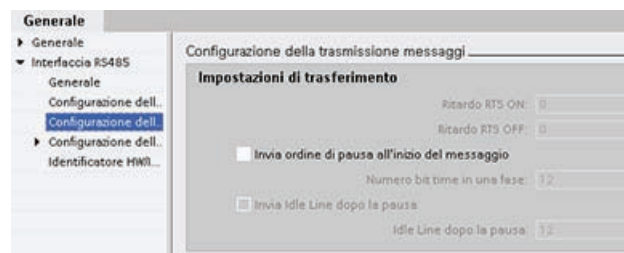
Quando il dispositivo di trasmissione riceve un carattere XOFF dal dispositivo ricevente, interrompe la trasmissione. La trasmissione riprende quando il dispositivo di trasmissione riceve un carattere XON. Se il dispositivo non riceve un carattere XON entro il tempo di attesa specificato nella configurazione della porta (Pagina 252), il CM annulla la trasmissione e segnala un errore al programma utente.

Il controllo del flusso software richiede una comunicazione full-duplex perché il partner di ricezione deve essere in grado di trasmettere XOFF al partner di trasmissione mentre è in corso la trasmissione. Il controllo del flusso software è utilizzabile solo nei messaggi che contengono unicamente caratteri ASCII. I protocolli binari non lo possono utilizzare.

8.4 Configurazione dei parametri di trasmissione (invio) e di ricezione

Affinché i PLC possano stabilire una comunicazione PtP è necessario configurare i parametri per la trasmissione (invio) e la ricezione di messaggi. Questi parametri determinano le modalità di comunicazione per la ricezione e la trasmissione dei messaggi da e verso un dispositivo di destinazione.

Configurazione dei parametri di trasmissione (invio)



Mentre si configura il CM si può definire il modo in cui un'interfaccia di comunicazione trasmette i dati specificando la proprietà "Configurazione della trasmissione messaggi" per il CM in questione.

È inoltre possibile configurare dinamicamente o modificare i parametri per la trasmissione dei messaggi inserendo nel programma utente l'istruzione SEND_CFG (Pagina 266).

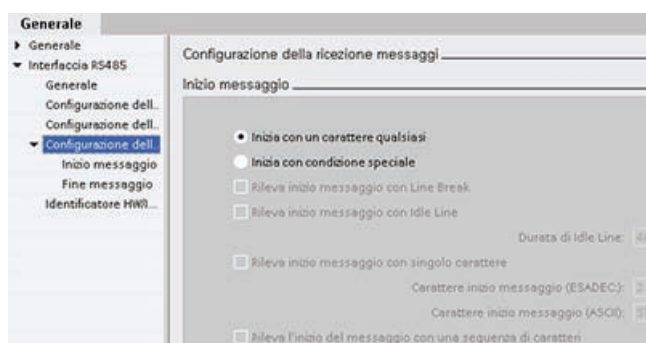
Nota

I valori dei parametri impostati dall'istruzione SEND_CFG nel programma utente sovrascrivono le impostazioni di configurazione della porta. In caso di mancanza dell'alimentazione la CPU non mantiene i parametri impostati con l'istruzione SEND_CFG.

Parametro	Definizione
Ritardo RTS ON	Specifica il tempo di attesa dopo l'attivazione dell'RTS prima dell'avvio della trasmissione. Il campo di valori valido è compreso tra 0 e 65535 ms, dove 0 è il valore di default. Questo parametro è valido soltanto se la configurazione della porta (Pagina 252) specifica il controllo del flusso "hardware". Il CTS viene valutato una volta trascorso l'intervallo di tempo Ritardo RTS ON. Questo parametro si applica solo ai moduli RS232.
Ritardo RTS OFF	Specifica il tempo di attesa prima della disattivazione dell'RTS dopo che la trasmissione è terminata. Il campo di valori valido è compreso tra 0 e 65535 ms, dove 0 è il valore di default. Questo parametro è valido soltanto se la configurazione della porta (Pagina 252) specifica il controllo del flusso "hardware". Questo parametro si applica solo ai moduli RS232.

Parametro	Definizione
Invia pausa all'inizio del messaggio Numero di Bit Time in una pausa	Specifica che all'avvio di ogni messaggio verrà inviato un break una volta trascorso l'intervallo di tempo Ritardo RTS ON (se configurato) e se CTS è attivo. È possibile specificare quanti tempi di bit costituiscono un break durante il quale la trasmissione viene interrotta. Il valore di default è 12 e il valore massimo è 65535, fino ad un limite di otto secondi.
Invia Idle Line dopo la pausa Idle Line dopo la pausa	Specifica che sarà inviata una "linea inattiva" dopo un break, all'inizio del messaggio. Il parametro "Idle line after a break" specifica quanti tempi di bit corrispondono a una "linea inattiva", dove la linea è mantenuta in una condizione di "mark". Il valore di default è 12 e il valore massimo è 65535, fino ad un limite di otto secondi.

Configurazione dei parametri di ricezione



Da Configurazione dispositivi è possibile impostare il modo in cui un'interfaccia di comunicazione riceve i dati e come riconosce l'inizio e la fine di un messaggio. Questi parametri vanno specificati in Configurazione della ricezione dei messaggi per il CM selezionato.

È anche possibile configurare dinamicamente o modificare i parametri della ricezione dei messaggi inserendo nel programma utente l'istruzione RCV_CFG (Pagina 267).

Nota

I valori dei parametri impostati dall'istruzione RCV_CFG nel programma utente sovrascrivono le impostazioni di configurazione della porta. In caso di mancanza dell'alimentazione la CPU non mantiene i parametri impostati con l'istruzione RCV_CFG.

Per maggiori informazioni consultare il paragrafo relativo all'istruzione RCV_CFG.

Parametri di inizio messaggio

È possibile determinare il modo in cui il modulo di comunicazione riconosce l'inizio di un messaggio. I caratteri di inizio e i caratteri del messaggio vengono memorizzati nel buffer di ricezione fino a quando non è soddisfatta una condizione di fine configurata.

È possibile specificare più di una condizione di inizio. Un messaggio viene considerato "avviato" solo se tutte le condizioni di avvio sono state soddisfatte. Se, ad esempio, si configura un timeout per linea inattiva e si specifica un carattere di inizio, il CM verifica prima se è presente la condizione di linea inattiva e poi cerca il carattere di inizio. Se vengono ricevuti altri caratteri (escluso quello specificato) il CM riavvia la ricerca dell'inizio del messaggio cercando nuovamente il timeout per linea inattiva.

8.4 Configurazione dei parametri di trasmissione (invio) e di ricezione

La priorità nella verifica delle condizioni di avvio è la seguente:

- Linea inattiva
- Interruzione di linea
- Caratteri o sequenze di caratteri

Se, mentre controlla le condizioni di avvio, il CM rileva una condizione non soddisfatta, riavvia il controllo dalla prima condizione richiesta.

Parametro	Definizione
Carattere inizio messaggio	La condizione "Carattere di inizio" indica che la ricezione di un particolare carattere determinerà l'inizio di un messaggio. Il carattere in questione sarà il primo del messaggio. Qualsiasi carattere ricevuto prima di questo viene eliminato.
Inizia con un carattere qualsiasi	La condizione "Qualsiasi carattere" specifica che la ricezione di qualsiasi carattere determinerà l'inizio di un messaggio. Il carattere in questione sarà il primo del messaggio.
Interruzione di linea	Le condizioni "Interruzione di linea" specificano che quando viene ricevuto un carattere di break venga avviata la ricezione del messaggio.
Linea inattiva	La condizione "Linea inattiva" specifica che la ricezione di un messaggio dovrebbe avere inizio quando la linea di ricezione è rimasta inattiva per il numero di tempi di bit specificato. Quando si verifica questa condizione il messaggio ha inizio.
Condizione speciale: Rileva inizio messaggio con singolo carattere	Specifica che un carattere particolare delimita l'inizio di un messaggio. Il valore di default è STX.
Condizione speciale: Rileva l'inizio del messaggio con una sequenza di caratteri	Indica che una particolare sequenza di caratteri delimita l'inizio di un messaggio. Per ogni sequenza è possibile specificare fino a cinque caratteri. Per ogni posizione è possibile indicare uno specifico carattere esadecimale oppure indicare che quel dato carattere deve essere ignorato nella sequenza. Le sequenze in arrivo vengono analizzate rispetto alle condizioni di inizio configurate, finché non viene soddisfatta la condizione di inizio. Quando la sequenza di inizio è soddisfatta inizia la valutazione della condizione di fine. È possibile configurare fino a cinque sequenze di caratteri specifiche che possono essere abilitate o disabilitate secondo necessità. La condizione di inizio è soddisfatta quando si verifica una qualsiasi delle sequenze di caratteri configurate.

Parametro	Definizione
Esempio di parametrizzazione	In questa configurazione la condizione di inizio è soddisfatta quando si verifica uno dei due seguenti casi: - Quando viene ricevuta una sequenza di cinque caratteri di cui il primo è 0x6A e il quinto è 0x1C. Le posizioni 2, 3 e 4 possono contenere qualsiasi carattere. Una volta ricevuto il quinto carattere inizia la valutazione delle condizioni di fine. - Quando vengono ricevuti due caratteri 0x6A consecutivi, preceduti da qualsiasi carattere. In questo caso la valutazione delle condizioni di fine inizia dopo la ricezione del secondo carattere 0x6A (3 caratteri). Il carattere che precede il primo 0x6A è incluso nella condizione di inizio. Esempi di sequenze che soddisfano questa condizione di inizio: <qualsiasi carattere> 6A 6A 6A 12 14 18 1C 6A 44 A5 D2 1C

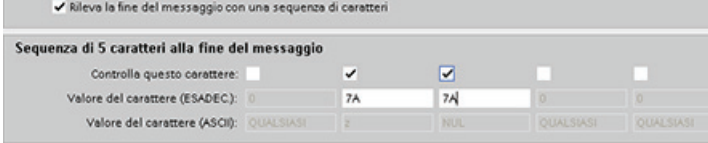
Parametri di fine messaggio

È anche possibile determinare il modo in cui l'interfaccia di comunicazione riconosce la fine di un messaggio. È possibile configurare più condizioni di fine messaggio. Se si verifica una delle condizioni configurate il messaggio termina.

È possibile specificare più condizioni contemporaneamente. Il messaggio termina quando viene soddisfatta una qualsiasi delle condizioni di fine. Ad esempio si può specificare una condizione di fine con un timeout di fine messaggio di 300 millisecondi, un timeout tra i caratteri di 40 tempi bit e una lunghezza massima di 50 byte. Il messaggio termina se la ricezione dura più di 300 millisecondi, se la distanza tra due caratteri supera i 40 tempi bit o dopo che sono stati ricevuti 50 byte.

Parametro	Definizione
Rileva la fine del messaggio mediante time out dello stesso	Il messaggio finisce trascorso il tempo di attesa configurato per la fine del messaggio stesso. Il periodo di timeout inizia quando viene ricevuto il primo carattere in base ai criteri di inizio messaggio. Il valore di default è 200 ms e il campo di valori va da 0 a 65535 ms.
Rileva la fine del messaggio mediante time out di risposta	Il messaggio finisce quando il tempo di attesa configurato per una risposta termina prima che sia stata ricevuta una sequenza di inizio valida. Il periodo di timeout per la risposta inizia quando termina una trasmissione. Il time out di risposta di default è di 200 ms e il campo va da 0 a 65535 ms. È necessario configurare un'altra condizione di fine per indicare l'effettiva fine di un messaggio.

8.4 Configurazione dei parametri di trasmissione (invio) e di ricezione

Parametro	Definizione
Rileva la fine del messaggio mediante gap intercaratteri	Il messaggio finisce trascorso il tempo di attesa massimo tra i caratteri consecutivi di un messaggio. Il valore di default per il gap tra i caratteri è 12 tempi di bit e il numero massimo è 65535 tempi di bit, fino ad un massimo di otto secondi.
Rileva la fine del messaggio sulla base della lunghezza max.	Il messaggio termina una volta ricevuto il numero massimo di caratteri configurato. Il valore di default è 0 byte e il valore massimo 1024 byte.
Rileva la lunghezza del messaggio	Il messaggio stesso specifica la propria lunghezza. Il messaggio termina una volta ricevuto un messaggio della lunghezza specificata. Il metodo per specificare e interpretare la lunghezza del messaggio è descritto più avanti.
Rileva la fine del messaggio con un carattere	Il messaggio finisce quando viene ricevuto un determinato carattere.
Rileva la fine del messaggio con una sequenza di caratteri	Il messaggio finisce quando viene ricevuta una determinata sequenza di caratteri. È possibile specificare una sequenza di cinque caratteri al massimo. Per ogni posizione è possibile indicare uno specifico carattere esadecimale oppure indicare che quel dato carattere deve essere ignorato nella sequenza. I caratteri introduttivi che vengono ignorati non fanno parte della condizione di fine. I caratteri conclusivi che vengono ignorati non fanno parte della condizione di fine.
Esempio di parametrizzazione	 In questo caso la condizione di fine è soddisfatta quando vengono ricevuti due caratteri 0x7A consecutivi, seguiti da due caratteri qualsiasi. Il carattere che precede la configurazione 0x7A 0x7A non fa parte della sequenza finale di caratteri. Per terminare la sequenza finale di caratteri sono necessari due caratteri dopo la configurazione 0x7A 0x7A. I valori dei caratteri nelle posizioni 4 e 5 sono irrilevanti, ma i caratteri devono essere ricevuti per soddisfare la condizione di fine.

Indicazione della lunghezza del messaggio nel messaggio stesso

Quando si seleziona la condizione speciale che stabilisce che il messaggio stesso specifichi la propria lunghezza, è necessario fornire tre parametri che contengono questa informazione.

La struttura effettiva del messaggio varia in base al protocollo in uso. I tre parametri necessari sono i seguenti:

- n: posizione del carattere (su base 1) all'interno del messaggio che inizia l'indicatore della lunghezza
- Valore lunghezza: numero di byte (uno, due o quattro) dell'indicatore della lunghezza
- Lunghezza m: numero di caratteri successivi all'indicatore della lunghezza e non inclusi nel conteggio della lunghezza

☒ Rileva la lunghezza dal messaggio

Offset del campo di lunghezza nel messaggio: 2 bytes

Dimensioni del campo di lunghezza: 1 bytes

Il campo di lunghezza che segue i dati non fa parte del messaggio: 0 bytes

☐ Rileva la fine del messaggio con una sequenza di caratteri

Questi campi compaiono in Configurazione della ricezione dei messaggi, nelle proprietà del dispositivo:

Esempio 1: si consideri un messaggio strutturato in base al seguente protocollo:

STX	Len (n)	Caratteri dal 3 al 14 contati in base alla lunghezza											
		ADR	PKE		INDEX		PWD		STW		HSW		BCC
1	2	3	4	5	6	7	8	9	10	11	12	13	14
STX	0x0C	xx	xxxx		xxxx		xxxx		xxxx		xxxx		xx

Configurare i parametri di lunghezza per la ricezione del messaggio come specificato di seguito:

- $n = 2$ (la lunghezza del messaggio inizia dal byte 2).
- Valore lunghezza = 1 (la lunghezza del messaggio è definita in un byte)
- Lunghezza $m = 0$ (dopo l'indicatore di lunghezza non ci sono altri caratteri esclusi dal conteggio della lunghezza. L'indicatore è seguito da dodici caratteri).

In questo esempio i caratteri dal 3 al 14 compreso sono i caratteri contati da Len (n).

Esempio 2: si consideri un altro messaggio strutturato in base al seguente protocollo:

SD1	Len (n)	Len (n)	SD2	Caratteri dal 5 al 10 contati in base alla lunghezza						FCS	ED
				DA	SA	FA	Unità dati=3 byte				
1	2	3	4	5	6	7	8	9	10	11	12
xx	0x06	0x06	xx	xx	xx	xx	xx	xx	xx	xx	xx

Configurare i parametri di lunghezza per la ricezione del messaggio come specificato di seguito:

- $n = 3$ (la lunghezza del messaggio inizia dal byte 3).
- Valore lunghezza = 1 (la lunghezza del messaggio è definita in un byte)
- Lunghezza $m = 3$ (dopo l'indicatore di lunghezza ci sono tre caratteri esclusi dal conteggio della lunghezza. Nel protocollo di questo esempio i caratteri SD2, FCS e ED non sono inclusi nel conteggio della lunghezza. Gli altri sei caratteri lo sono, pertanto il numero complessivo di caratteri che seguono l'indicatore della lunghezza è nove).

In questo esempio i caratteri dal 5 al 10 compreso sono i caratteri contati da Len (n).

8.5 Programmazione della comunicazione PtP

STEP 7 Basic mette a disposizione istruzioni avanzate che consentono al programma utente di eseguire la comunicazione punto a punto con il protocollo che vi è stato progettato e specificato. Le istruzioni possono essere classificate in due categorie:

- Istruzioni di configurazione
- Istruzioni di comunicazione

Istruzioni di configurazione

Perché il programma utente possa avviare una comunicazione PtP è necessario configurare la porta di comunicazione e i parametri per l'invio e la ricezione dei dati.

Per configurare la porta e i messaggi per ciascun modulo di comunicazione si può utilizzare Configurazione dispositivi o eseguire le seguenti istruzioni del programma utente:

- PORT_CFG
- SEND_CFG
- RCV_CFG

Istruzioni di comunicazione

Le istruzioni di comunicazione PtP consentono al programma utente di inviare e ricevere messaggi dai moduli di comunicazione. Per informazioni sul trasferimento dei dati mediante queste istruzioni consultare il paragrafo sulla coerenza dei dati (Pagina 94).

Tutte le funzioni PtP funzionano in modo asincrono. Il programma utente può utilizzare un'architettura di interrogazione per determinare lo stato delle trasmissioni e delle ricezioni. Le istruzioni SEND_PTP e RCV_PTP possono essere eseguite simultaneamente. I moduli di comunicazione memorizzano i messaggi di trasmissione e ricezione in base alle necessità, fino alla dimensione massima del buffer pari a 1024 byte.

I moduli di comunicazione inviano e ricevono messaggi dai dispositivi PtP. Il protocollo per i messaggi è contenuto in un buffer ricevuto o trasmesso da una specifica porta di comunicazione.

Ulteriori istruzioni consentono di resettare il buffer di ricezione e di ricevere e impostare specifici segnali RS232.

- SEND_PTP
- RCV_PTP
- RCV_RST
- SGN_GET
- SGN_SET

8.5.1 Architettura di interrogazione

Le istruzioni punto a punto dell'S7-1200 devono essere richiamate ciclicamente/periodicamente allo scopo di controllare se sono stati ricevuti messaggi. Interrogando la trasmissione il programma utente riesce a rilevare quando la trasmissione è terminata.

Architettura di interrogazione: master

La tipica sequenza di un master è la seguente:

1. Un'istruzione SEND_PTP avvia una trasmissione al modulo di comunicazione.
2. L'istruzione SEND_PTP viene eseguita in cicli successivi perché interroghi lo stato "trasmissione conclusa".
3. Quando SEND_PTP indica che la trasmissione è terminata il codice utente può prepararsi a ricevere la risposta.
4. L'istruzione RCV_PTP viene eseguita ripetutamente perché controlli se viene ricevuta una risposta. Se il CM rileva un messaggio di risposta, RCV_PTP lo copia nella CPU e indica che sono stati ricevuti nuovi dati.

5. Il programma utente può elaborare la risposta.
6. Tornare alla prima operazione e ripetere il ciclo.

Architettura di interrogazione: slave

La tipica sequenza di uno slave è la seguente:

1. Il programma utente deve eseguire l'istruzione RCV_PTP in tutti i cicli di scansione.
2. Quando il CM riceve una richiesta l'istruzione RCV_PTP indica che sono stati ricevuti nuovi dati e copia la richiesta nella CPU.
3. Il programma utente deve elaborare la richiesta e generare una risposta.
4. Trasmettere la risposta al master con un'istruzione SEND_PTP.
5. Eseguire SEND_PTP ripetutamente per assicurarsi che la trasmissione venga effettuata.
6. Tornare alla prima operazione e ripetere il ciclo.

Lo slave deve richiamare RCV_PTP abbastanza frequentemente da ricevere una trasmissione dal master prima che quest'ultimo vada in timeout nell'attesa di una risposta. A tal fine il programma utente può richiamare RCV_PTP da un OB di ciclo il cui tempo di ciclo è sufficiente per ricevere una trasmissione dal master prima che termini il periodo di timeout. Se si imposta il tempo di ciclo per l'OB in modo tale da garantire due esecuzioni entro il periodo di timeout del master, il programma utente dovrebbe ricevere le trasmissioni senza perderne alcuna.

8.6 Istruzioni punto a punto (PtP)

8.6.1 Parametri comuni delle istruzioni punto a punto

Segnalazioni dei LED del modulo di comunicazione

Sul modulo di comunicazione (CM) sono presenti tre indicatori LED:

- LED di diagnostica: Il LED di diagnostica emette una luce rossa lampeggiante finché non viene indirizzato dalla CPU. Dopo l'accensione della CPU, il LED controlla i moduli e indirizza il modulo CM. Il LED di diagnostica inizia ad emettere una luce verde lampeggiante. Ciò significa che la CPU ha indirizzato il CM ma non lo ha ancora configurato. La configurazione viene caricata nel modulo quando si carica il programma nella CPU. Dopo il caricamento nella CPU, il LED di diagnostica del modulo di comunicazione dovrebbe accendersi con una luce verde fissa.
- LED di trasmissione: Il LED di trasmissione si trova sopra al LED di ricezione. Si accende quando è in corso la trasmissione dei dati attraverso la porta di comunicazione.
- LED di ricezione: Questo LED si accende mentre la porta di comunicazione riceve i dati.

Risoluzione dei tempi di bit

Per diversi parametri si indica il numero di tempi di bit alla velocità di trasmissione configurata. Specificando il parametro in tempi di bit lo si rende indipendente dalla velocità di trasmissione. I parametri che vengono espressi in tempi di bit possono assumere il valore massimo di 65535. Tuttavia il tempo massimo che può essere misurato da un S7-1200 è di 8 secondi.

Parametro di ingresso REQ

Molte delle istruzioni punto a punto (PtP) usano un ingresso REQ che avvia l'operazione in una transazione da low a high (fronte di salita). L'ingresso REQ deve essere high (vero) durante l'esecuzione di un'istruzione e può restare vero senza limitazione di tempo. L'istruzione non avvia un'altra operazione fino a quando non viene richiamata con l'ingresso REQ falso in modo da poter resettare lo stato dell'ingresso REQ. Questo affinché l'istruzione possa identificare la transizione da low a high e avviare l'operazione successiva.

Quando si inserisce un'istruzione PtP viene richiesto di identificare il DB di istanza. Usare un DB unico per ogni tipo di istruzione PtP. Tutte le istruzioni SEND_PTP per una data porta devono infatti avere lo stesso DB di istanza, mentre le istruzioni SEND_PTP e RCV_PTP devono avere DB diversi. Questo assicura che tutte le istruzioni gestiscano correttamente gli ingressi quali REQ.

Parametro di ingresso PORT

Dall'elenco a discesa (associato all'ingresso PORT) selezionare l'identificativo della porta per il CM che deve essere gestito da questa istanza dell'istruzione. Il numero è disponibile anche come "identificativo hardware" nelle informazioni di configurazione del CM.

Parametri di uscita NDR, DONE, ERROR e STATUS

- L'uscita DONE indica che l'operazione richiesta si è conclusa senza errori. L'uscita resta impostata per un ciclo di scansione.
- L'uscita NDR (New Data Ready - nuovi dati disponibili) indica che l'azione richiesta si è conclusa senza errori e che sono stati ricevuti nuovi dati. L'uscita resta impostata per un ciclo di scansione.
- L'uscita ERROR indica che l'azione richiesta si è conclusa con un errore. L'uscita resta impostata per un ciclo di scansione.
- L'uscita STATUS si usa per segnalare errori o risultati di stato intermedi.
 - Se è impostato il bit DONE o NDR, STATUS sarà impostato a 0 o su un codice di informazione.
 - Se è impostato il bit ERROR, STATUS sarà impostato su un codice di errore.
 - Se non è impostato nessuno dei bit precedenti, l'istruzione restituisce risultati che descrivono lo stato attuale della funzione, ad es. lo stato "occupato".

Codici comuni delle condizioni

STATUS (W#16#....)	Descrizione
0000	Nessun errore
8x3A	Puntatore non ammesso nel parametro x
8070	Tutta la memoria di istanza interna è stata utilizzata
8080	Numero di porta non ammesso
8081	Timeout, errore del modulo o altro errore interno
8082	La parametrizzazione non è andata a buon fine perché è già in corso in background
8083	Overflow del buffer: il CM ha restituito un messaggio ricevuto che aveva una lunghezza maggiore rispetto a quella consentita dal relativo parametro.
8090	Lunghezza messaggio errata, sottomodulo errato o messaggio non ammesso
8091	Versione errata nel messaggio di parametrizzazione
8092	Lunghezza del record errata nel messaggio di parametrizzazione

8.6.2 Istruzione PORT_CFG



L'istruzione PORT_CFG (Configurazione della porta) consente di modificare dal programma alcuni parametri della porta, ad es. la velocità di trasmissione.

Impostare la configurazione statica iniziale della porta nelle proprietà di Configurazione dispositivi o utilizzare i valori di default. È possibile eseguire l'istruzione PORT_CFG dal programma utente per modificare la configurazione. Le modifiche apportate alla configurazione con PORT_CFG non vengono memorizzate in modo permanente nella CPU. I parametri configurati nel dispositivo di configurazione vengono ripristinati quando la CPU passa da RUN a STOP e dopo uno spegnimento/riaccensione.

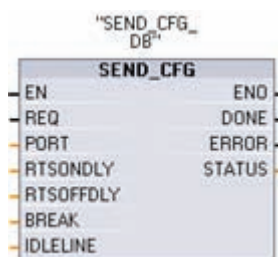
Per maggiori informazioni consultare i paragrafi Configurazione delle porte di comunicazione (Pagina 252) e Gestione del controllo di flusso (Pagina 253).

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Attiva la modifica alla configurazione in seguito a un fronte di salita di questo ingresso.
PORT	IN	PORT	Identificativo della porta di comunicazione: questo indirizzo logico è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
PROTOCOL	IN	UInt	0 - Protocollo per la comunicazione punto a punto 1..n - Definizione futura di protocolli specifici
BAUD	IN	UInt	Velocità di trasmissione della porta: 1 - 300 baud 2 - 600 baud 3 - 1200 baud 4 - 2400 baud 5 - 4800 baud 6 - 9600 baud 7 - 19200 baud 8 - 38400 baud 9 - 57600 baud 10 - 76800 baud 11 - 115200 baud
PARITY	IN	UInt	Parità della porta: 1 - Nessuna parità 2 - Parità pari 3 - Parità dispari 4 - Parità mark 5 - Parità space
DATABITS	IN	UInt	Bit per carattere: 1 - 8 bit di dati 2 - 7 bit di dati
STOPBITS	IN	UInt	Bit di stop: 1 - 1 bit di stop 2 - 2 bit di stop
FLOWCTRL	IN	UInt	Controllo del flusso: 1 - Nessun controllo del flusso 2 - XON/XOFF 3 - Hardware RTS sempre ON 4 - Hardware RTS sempre disattivato
XONCHAR	IN	Char	Specificare il carattere usato come XON. Si tratta tipicamente di un carattere DC1 (11H). Questo parametro viene valutato soltanto se è abilitato il controllo del flusso.
XOFFCHAR	IN	Char	Specificare il carattere usato come XOFF. Si tratta tipicamente di un carattere DC3 (13H). Questo parametro viene valutato soltanto se è abilitato il controllo del flusso.
XWAITIME	IN	UInt	Specificare quanto si deve attendere un carattere XON dopo la ricezione di un carattere XOFF oppure quanto si deve attendere il segnale CTS dopo avere abilitato RTC (da 0 a 65535 ms). Questo parametro viene valutato soltanto se è abilitato il controllo del flusso.
DONE	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa senza errori
ERROR	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa con un errore
STATUS	OUT	WORD	Codice della condizione di esecuzione

STATUS (W#16#....)	Descrizione
80A0	Il protocollo specificato non esiste.
80A1	La velocità di trasmissione specificata non esiste.
80A2	L'opzione di parità specificata non esiste.
80A3	Il numero di bit di dati specificato non esiste.
80A4	Il numero di bit di stop specificato non esiste.
80A5	Il tipo di controllo del flusso specificato non esiste.
80A6	Il tempo di attesa è 0 e il controllo del flusso è attivo.
80A7	XON e XOFF sono valori non ammessi.

8.6.3 Istruzione SEND_CFG



L'istruzione SEND_CFG (Configura trasmissione) consente di configurare in modo dinamico i parametri per la trasmissione seriale in una porta di comunicazione punto a punto. Non appena viene eseguita SEND_CFG tutti i messaggi in coda in un modulo di comunicazione (CM) vengono eliminati.

Impostare la configurazione statica iniziale della porta nelle proprietà di Configurazione dispositivi o utilizzare i valori di default. È possibile eseguire l'istruzione SEND_CFG dal programma utente per modificare la configurazione. Le modifiche alla configurazione di SEND_CFG non vengono memorizzate in modo permanente nel PLC. I parametri configurati nel dispositivo di configurazione vengono ripristinati quando la CPU passa da RUN a STOP e dopo uno spegnimento/riaccensione. Consultare Configurazione dei parametri di trasmissione (invio) e di ricezione (Pagina 255).

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Attiva la modifica alla configurazione in seguito a un fronte di salita di questo ingresso.
PORT	IN	PORT	Identificativo della porta di comunicazione: questo indirizzo logico è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.
RTSONDLY	IN	UInt	Millisecondi di attesa dopo l'abilitazione dell'RTS prima che si verifichi una trasmissione di dati Tx. questo parametro è valido solo se è abilitato il controllo del flusso. Da 0 a 65535 ms. 0 disattiva la funzione.
RTSOFFDLY	IN	UInt	Millisecondi di attesa dopo la trasmissione dei dati Tx prima della disabilitazione di RTS: questo parametro è valido solo se è abilitato il controllo del flusso. Da 0 a 65535 ms. 0 disattiva la funzione.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
BREAK	IN	UInt	Questo parametro specifica che all'inizio di ogni messaggio viene inviato un break per il numero specificato di tempi di bit. Il valore massimo è di 65535 tempi di bit. 0 disattiva la funzione. 8 secondi al massimo.
IDLELINE	IN	UInt	Questo parametro specifica che la linea resta inattiva per il numero specificato di tempi di bit prima dell'inizio di ogni messaggio. Il valore massimo è di 65535 tempi di bit. 0 disattiva la funzione. 8 secondi al massimo.
DONE	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa senza errori
ERROR	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa con un errore
STATUS	OUT	WORD	Codice della condizione di esecuzione

STATUS (W#16#....)	Descrizione
80B0	Configurazione dell'allarme di trasmissione non ammessa
80B1	La durata del break supera il valore consentito (2500 tempi di bit)
80B2	Il tempo di inattività supera il valore consentito (2500 tempi di bit)

8.6.4 Istruzione RCV_CFG



L'istruzione RCV_CFG (Configura ricezione) consente di configurare in modo dinamico i parametri per la ricezione seriale in una porta di comunicazione PtP. L'istruzione configura le condizioni che segnalano l'inizio e la fine dei messaggi ricevuti. Non appena viene eseguita RCV_CFG tutti i messaggi in coda in un CM vengono eliminati.

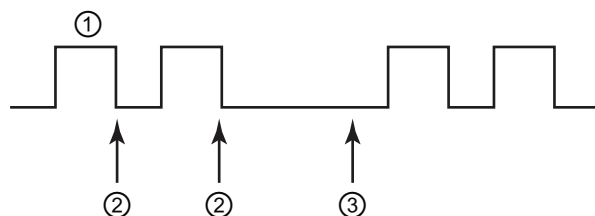
Impostare la configurazione statica iniziale della porta del CM nelle proprietà di Configurazione dispositivi o utilizzare i valori di default. È possibile eseguire l'istruzione RCV_CFG dal programma utente per modificare la configurazione. Le modifiche alla configurazione di RCV_CFG non vengono memorizzate in modo permanente nel PLC. I parametri configurati nel dispositivo di configurazione vengono ripristinati quando la CPU passa da RUN a STOP e dopo uno spegnimento/riaccensione. Per maggiori informazioni consultare Configurazione dei parametri di ricezione (Pagina 255).

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Attiva la modifica alla configurazione in seguito a un fronte di salita di questo ingresso.
PORT	IN	PORT	Identificativo della porta di comunicazione: questo indirizzo logico è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.
CONDITIONS	IN	CONDITIONS	La struttura dei dati di CONDITIONS specifica le condizioni di inizio e fine del messaggio descritte di seguito.
DONE	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa senza errori
ERROR	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa con un errore
STATUS	OUT	WORD	Codice della condizione di esecuzione

Condizioni di inizio per l'istruzione RCV_PTP

L'istruzione RCV_PTP usa la configurazione specificata dall'istruzione RCV_CFG per determinare l'inizio e la fine dei messaggi di comunicazione punto a punto. L'inizio di un messaggio è determinato dalle relative condizioni. L'inizio di un messaggio può essere determinato da un'unica condizione di inizio o da una combinazione di condizioni. Se sono state specificate più condizioni di avvio, il messaggio viene avviato quando sono tutte soddisfatte. Possibili condizioni di inizio:

- Il carattere di inizio indica che la ricezione di un particolare carattere determinerà l'inizio di un messaggio. Il carattere in questione sarà il primo del messaggio. Qualsiasi carattere ricevuto prima di questo viene eliminato.
- "Qualsiasi carattere" specifica che la ricezione di qualsiasi carattere determinerà l'inizio di un messaggio. Il carattere in questione sarà il primo del messaggio.
- "Interruzione di linea" specifica che quando viene ricevuto un carattere di break venga avviata la ricezione del messaggio.
- "Linea inattiva" specifica che la ricezione di un messaggio dovrebbe avere inizio quando la linea di ricezione è rimasta inattiva per il numero di tempi di bit specificato. Quando si verifica questa condizione il messaggio ha inizio.



- ① Caratteri
- ② Riavvia il temporizzatore di linea inattiva
- ③ Viene rilevata la condizione di linea inattiva e avviata la ricezione del messaggio
- Sequenze variabili: È possibile creare delle condizioni di inizio basate su un numero variabile di sequenze di caratteri (fino ad un massimo di 4) composte da un numero

variabile di caratteri (fino a un massimo di 5). Per ogni posizione all'interno di ciascuna sequenza si può selezionare un carattere specifico o un carattere jolly, ovvero qualsiasi carattere. Questa condizione di inizio può essere usata quando diverse sequenze di caratteri indicano l'inizio di un messaggio.

Si consideri il seguente messaggio ricevuto con codifica esadecimale: "68 10 aa 68 bb 10 aa 16" e le sequenze di inizio configurate mostrate nella tabella più sotto. Le sequenze di inizio cominciano ad essere valutate dopo la ricezione riuscita del primo carattere 68H. Dopo la ricezione del quarto carattere (il secondo 68H) la condizione di inizio 1 è soddisfatta. Una volta soddisfatte le condizioni di inizio comincia la valutazione di quelle di fine.

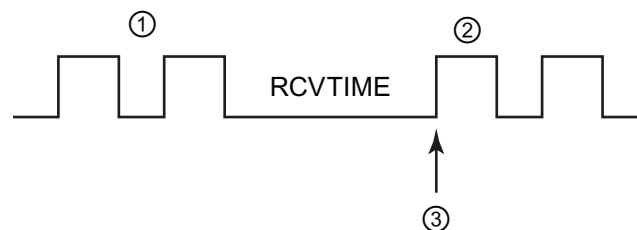
L'elaborazione della sequenza di inizio può essere interrotta in seguito a diversi errori di parità, di framing o di temporizzazione intercaratteri. In seguito a questi errori i messaggi non vengono ricevuti perché non viene soddisfatta la condizione di inizio.

Condizione di inizio	Primo carattere	Primo carattere +1	Primo carattere +2	Primo carattere +3	Primo carattere +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Condizioni di fine per l'istruzione RCV_PTP

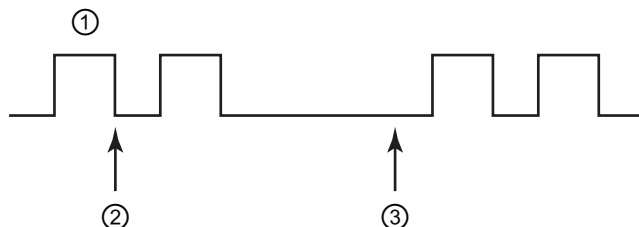
Per determinare la fine di un messaggio se ne devono specificare le condizioni di fine. Quando queste si verificano il messaggio viene concluso. Possibili condizioni di fine messaggio:

- "Timeout della risposta" indica che un carattere della risposta deve essere ricevuto entro il tempo specificato da RCVTIME. Il timer si avvia appena termina la trasmissione e il modulo inizia a ricevere il messaggio. Se non viene ricevuto alcun carattere entro il periodo RCVTIME, viene segnalato un errore alla relativa istruzione RCV_PTP. Il timeout della risposta non definisce una specifica condizione di fine, ma indica soltanto che deve essere ricevuto un carattere entro il tempo specificato. Per definire la condizione di inizio e di fine dei messaggi di risposta si deve utilizzare una condizione di fine a parte.



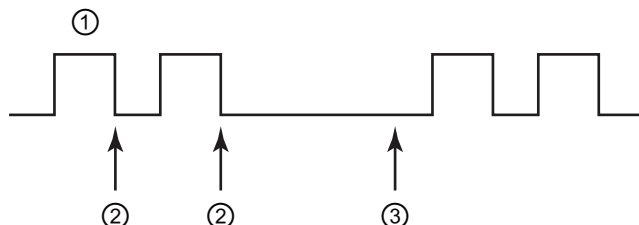
- ① Caratteri trasmessi
- ② Caratteri ricevuti
- ③ Il primo carattere deve essere ricevuto entro il tempo indicato

- "Timeout del messaggio" specifica che un messaggio deve essere ricevuto entro il tempo specificato da MSGTIME. Il temporizzatore si avvia non appena viene soddisfatta la condizione di inizio specificata.



- ① Caratteri ricevuti
- ② Condizione di inizio messaggio soddisfatta: viene avviato il temporizzatore dei messaggi
- ③ Il temporizzatore dei messaggi raggiunge il valore previsto e interrompe il messaggio

- "Tempo intercaratteri" è il tempo che trascorre tra la fine di un carattere (l'ultimo bit di stop) e la fine del carattere successivo. Se il tempo tra due caratteri supera il numero di tempi di bit configurati, il messaggio viene concluso.



- ① Caratteri ricevuti
- ② Riavvia il temporizzatore intercaratteri
- ③ Il temporizzatore intercaratteri raggiunge il valore previsto e interrompe il messaggio segnalando degli errori

- Lunghezza massima: Una volta ricevuto il numero di caratteri specificato l'operazione di ricezione si arresta. Questa condizione può essere utilizzata per impedire un errore di overrun del buffer dei messaggi.

Se questa condizione di fine è abbinata a condizioni di fine con timeout e se si verifica una di queste condizioni, vengono emessi tutti i caratteri validi ricevuti, anche se non è stata raggiunta la lunghezza massima. Questo consente di supportare protocolli di lunghezza diversa anche se è nota soltanto la lunghezza massima.

- Condizione combinata "N + valore lunghezza + lunghezza M". Questa condizione di fine può essere usata per elaborare un messaggio di lunghezza variabile che contiene un campo per la lunghezza.
 - N specifica la posizione (numero di caratteri nel messaggio) da cui inizia il campo della lunghezza. (base 1)
 - "Valore lunghezza" specifica le dimensioni del campo della lunghezza. Sono valori validi 1, 2 o 4 byte.
 - Il parametro "Lunghezza M" specifica il numero dei caratteri di fine (successivi al campo della lunghezza) non compresi nella lunghezza del messaggio. Questo può essere usato per specificare la lunghezza di un campo per la somma di controllo le cui dimensioni non sono comprese nel campo della lunghezza.
 - Sempre a titolo esemplificativo, si può considerare un formato di messaggio composto da un carattere di inizio, un carattere di indirizzo, un campo della lunghezza di un

byte, i dati del messaggio, i caratteri della somma di controllo e un carattere di fine. Le voci contrassegnate con "Len" corrispondono al parametro N. Il valore di N è 3, a indicare che il byte della lunghezza si trova nella posizione 3 dall'inizio del messaggio. Il valore del parametro "Valore lunghezza" è 1, a indicare che il valore di lunghezza del messaggio è contenuto in 1 byte. La somma di controllo e i campi dei caratteri di fine corrispondono al parametro "Lunghezza M". Il valore del parametro "Lunghezza M" è 3 e specifica che il numero di byte dei campi per la somma di controllo e i caratteri.

Car. di inizio (1)	Indirizzo (2)	Len (N) (3)	Messaggio ... (x)		Somma di controllo e car. di fine Lunghezza M x+1 x+2 x+3		
xx	xx	xx	xx	xx	xx	xx	xx

- **Caratteri variabili:** Questa condizione di fine può essere usata per concludere la ricezione in base a diverse sequenze di caratteri che possono essere composte da un numero di caratteri variabile (fino ad un massimo di 5). Per ogni posizione all'interno di ciascuna sequenza si può selezionare un carattere specifico o un carattere jolly, a indicare che la condizione può essere soddisfatta da qualsiasi carattere. I caratteri introduttivi configurati per essere ignorati non devono necessariamente far parte del messaggio. I caratteri conclusivi che vengono ignorati devono essere invece inclusi nel messaggio.

Struttura del tipo di dati del parametro CONDITIONS, parte 1 (condizioni di inizio)

Parametro	Tipo di parametro	Tipo di dati	Descrizione
STARTCOND	IN	UInt	Specifica la condizione di inizio: • 01H - Carattere di inizio • 02H - Qualsiasi carattere • 04H - Interruzione di linea • 08H - Linea inattiva • 10H - Sequenza 1 • 20H - Sequenza 2 • 40H - Sequenza 3 • 80H - Sequenza 4
IDLETIME	IN	UInt	Numero di tempi di bit richiesto per il timeout di linea inattiva. Usato soltanto con una condizione di linea inattiva. Da 0 a 65535.
STARTCHAR	IN	BYTE	Carattere di inizio usato con la condizione "carattere di inizio".

Parametro	Tipo di parametro	Tipo di dati	Descrizione
STRSEQ1CTL	IN	BYTE	Sequenza 1, comando ignora/confronta per ogni carattere, ovvero i bit di attivazione per ogni carattere della sequenza di inizio. • 01H - Carattere 1 • 02H - Carattere 2 • 04H - Carattere 3 • 08H - Carattere 4 • 10H - Carattere 5 Se si disattiva il bit associato ad un carattere, qualsiasi carattere che occupa la stessa posizione all'interno della sequenza rappresenta una corrispondenza.
STRSEQ1	IN	Char[5]	Sequenza 1, caratteri di inizio (5 caratteri)
STRSEQ2CTL	IN	BYTE	Sequenza 2, comando ignora/confronta per ogni carattere
STRSEQ2	IN	Char[5]	Sequenza 2, caratteri di inizio (5 caratteri)
STRSEQ3CTL	IN	BYTE	Sequenza 3, comando ignora/confronta per ogni carattere
STRSEQ3	IN	Char[5]	Sequenza 3, caratteri di inizio (5 caratteri)
STRSEQ4CTL	IN	BYTE	Sequenza 4, comando ignora/confronta per ogni carattere
STRSEQ4	IN	Char[5]	Sequenza 4, caratteri di inizio (5 caratteri)

Struttura del tipo di dati del parametro CONDITIONS, parte 2 (condizioni di fine)

Parametro	Tipo di parametro	Tipo di dati	Descrizione
ENDCOND	IN	UInt	Questo parametro specifica la condizione di fine del messaggio: • 01H - Tempo di risposta • 02H - Durata del messaggio • 04H - Gap intercaratteri • 08H - Lunghezza massima • 10H - N + LEN + M • 20H - Sequenza
MAXLEN	IN	UInt	Lunghezza massima del messaggio: usata solo se è selezionata la condizione di fine "lunghezza massima". Da 0 a 1023 byte.
N	IN	UInt	Posizione di byte del campo della lunghezza all'interno del messaggio. Usata solo se è impostata la condizione di fine N + LEN + M. Da 1 a 1023 byte.
LENGTHSIZE	IN	UInt	Dimensioni del campo di byte (1, 2 o 4 byte). Usata solo se è impostata la condizione di fine N + LEN + M.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
LENGTHM	IN	UInt	Specificare il numero di caratteri successivi al campo della lunghezza e non compresi nel valore della lunghezza. Usata solo se è impostata la condizione di fine $N + LEN + M$. Da 0 a 255 byte.
RCVTIME	IN	UInt	Specificare quanto si deve attendere per la ricezione del primo carattere. Se entro il tempo specificato non viene ricevuto alcun carattere la ricezione viene conclusa con un errore. Questo parametro si usa soltanto se è impostata la condizione del tempo di risposta. Da 0 a 65535 tempi di bit, max. 8 secondi. Questo parametro non viene realmente valutato come una condizione di fine perché valuta soltanto condizioni di inizio. Si deve quindi selezionare una condizione di fine specifica.
MSGTIME	IN	UInt	Specificare quanto si deve attendere per la ricezione dell'intero messaggio dopo che è stato ricevuto il primo carattere. Questo parametro si usa soltanto quando è selezionata la condizione di timeout del messaggio. Da 0 a 65535 millisecondi.
CHARGAP	IN	UInt	Specificare il numero di tempi di bit tra i caratteri. Se il numero di tempi di bit tra i caratteri supera il valore specificato, la condizione di fine è soddisfatta. Questo parametro si usa soltanto se è impostata la condizione del gap intercaratteri. Da 0 a 65535 millisecondi.
ENDSEQ1CTL	IN	BYTE	Sequenza 1, comando ignora/confronta per ogni carattere, ovvero i bit di attivazione per ogni carattere della sequenza di fine. Il carattere 1 è il bit 0, il carattere 2 è il bit 1, ..., il carattere 5 è il bit 4. Se si disattiva il bit associato ad un carattere, qualsiasi carattere che occupa la stessa posizione all'interno della sequenza rappresenta una corrispondenza.
ENDSEQ1	IN	Char[5]	Sequenza 1, caratteri di inizio (5 caratteri)

Codici delle condizioni di errore

STATUS (W#16#....)	Descrizione
80C0	È stata selezionata una condizione di inizio non ammessa
80C1	È stata selezionata una condizione di fine non ammessa, non è stata selezionata alcuna condizione di fine
80C2	È stato attivato un allarme di ricezione e questo non è possibile
80C3	È stata abilitata la condizione di fine Lunghezza massima e la lunghezza massima è $0 > 1024$
80C4	La lunghezza calcolata è stata abilitata e il valore N è ≥ 1023
80C5	La lunghezza calcolata è stata abilitata e la lunghezza non è 1, 2 o 4
80C6	La lunghezza calcolata è stata abilitata e il valore M è > 255
80C7	La lunghezza calcolata è stata abilitata ed è > 1024

STATUS (W#16#....)	Descrizione
80C8	Il timeout della risposta è stato abilitato ed è pari a zero
80C9	Il timeout del gap intercaratteri è stato abilitato ed è pari a zero o > 2500
80CA	Il timeout di linea inattiva è stato abilitato ed è pari a zero o > 2500
80CB	La sequenza di fine è stata abilitata ma tutti i caratteri sono "don't care"
80CC	La sequenza di inizio (una qualsiasi di 4) è stata abilitata ma tutti i caratteri sono "don't care"

8.6.5 Istruzione SEND_PTP



L'istruzione SEND_PTP (Trasmetti dati punto a punto) avvia la trasmissione dei dati. SEND_PTP trasferisce il buffer specificato nel CM. Il programma della CPU continua mentre il CM invia i dati alla velocità di trasmissione specificata. Può essere attiva una sola operazione di trasmissione per volta. Il CM restituisce un errore se viene eseguita una seconda SEND_PTP mentre sta già trasmettendo un messaggio.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Attiva la trasmissione richiesta in seguito a un fronte di salita in questo ingresso di abilitazione della trasmissione. Viene così avviato il trasferimento dei contenuti del buffer nel modulo di comunicazione (CM) punto a punto.
PORT	IN	PORT	Identificativo della porta di comunicazione: questo indirizzo logico è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.
BUFFER	IN	Versione	Questo parametro punta all'indirizzo iniziale del buffer di trasmissione. I dati e gli array booleani non sono supportati.
LENGTH	IN	UInt	Lunghezza di frame trasmessa da byte. Per la trasmissione di una struttura complessa utilizzare sempre una lunghezza pari a 0.
PTRCL	IN	Bool	Questo parametro seleziona il buffer per la normale comunicazione punto a punto o per protocolli specifici forniti da Siemens, implementati nel CM collegato. Falso = operazioni punto a punto comandate dal programma utente. (unica opzione valida)
DONE	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa senza errori

Parametro	Tipo di parametro	Tipo di dati	Descrizione
ERROR	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa con un errore
STATUS	OUT	WORD	Codice della condizione di esecuzione

Mentre è in corso una trasmissione, le uscite DONE e ERROR sono impostate su "falso". Quando l'operazione di trasmissione è conclusa, l'uscita DONE o l'uscita ERROR saranno impostate su "vero" per un ciclo di scansione, per indicare lo stato della trasmissione. Quando DONE o ERROR sono impostate su "vero" l'uscita STATUS è valida.

Se il modulo di comunicazione (CM) accetta i dati di trasmissione, l'istruzione restituisce lo stato 16#7001. Le esecuzioni successive di SEND_PTP restituiscono un 16#7002 se il CM è ancora impegnato a trasmettere. Al termine della trasmissione, se non si è verificato alcun errore il CM restituisce lo stato 16#0000. Le esecuzioni successive di SEND_PTP con REQ low restituiscono lo stato 16#7000 (non occupato).

Rapporto tra i valori di uscita e REQ:

Si presuppone che l'istruzione sia richiamata periodicamente per controllare lo stato della trasmissione. Nel seguente schema si presuppone che l'istruzione venga richiamata ad ogni ciclo di scansione (rappresentato dai valori STATUS).

REQ							
DONE							
ERROR							
STATUS	7000H	7001H	7002H	7002H	7002H	0000H	7000H

Il seguente schema mostra come i parametri DONE e STATUS sono validi per una sola scansione se sulla linea REQ è presente un impulso (per una scansione) per avviare la trasmissione.

REQ								
DONE								
ERROR								
STATUS	7000H	7001H	7002H	7002H	7002H	0000H	7000H	7000H

Il seguente schema mostra il rapporto tra i parametri DONE, ERROR e STATUS in caso di errore.

REQ								
DONE								
ERROR								
STATUS	7000H	7001H	7002H	7002H	7002H	80D1H	7000H	7000H

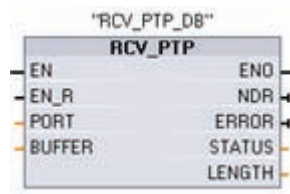
STATUS (W#16#....)	Descrizione
80D0	Nuova richiesta con trasmettitore attivo
80D1	Trasmissione annullata perché non è pervenuto alcun CTS entro il tempo di attesa
80D2	Trasmissione annullata perché non è pervenuto alcun DSR dal dispositivo DCE
80D3	Trasmissione annullata a causa di un overflow della coda d'attesa (trasmissione di più di 1024 byte)
7000	Libero
7001	Occupato se accetta la richiesta (prima chiamata)
7002	Occupato su un'interrogazione (n-sima chiamata)

Interazione dei parametri LENGTH e DATA di PTP_SEND

La dimensione minima dei dati trasmissibili con l'istruzione PTP_SEND è di un byte. Il parametro DATA determina la dimensione dei dati da trasmettere. Non accetta dati o array booleani.

Parametro LENGTH	Parametro DATA	Descrizione
LENGTH = 0	Non utilizzato	I dati completi vengono inviati come definito al parametro DATA. Non occorre specificare il numero di byte trasmessi se LENGTH = 0.
LENGTH > 0	Tipo di dati semplice	Il valore LENGTH deve contenere il numero di byte di questo tipo di dati. In caso contrario il trasferimento non avviene e viene restituito l'errore 8088H.
	Struttura	Il valore LENGTH può contenere un numero di byte inferiore alla lunghezza di byte complessiva della struttura. In questo caso vengono trasferiti solo i primi byte LENGTH.
	Array	Il valore LENGTH può contenere un numero di byte inferiore alla lunghezza di byte complessiva dell'array. In questo caso vengono trasferiti solo gli elementi dell'array che rientrano completamente nei byte LENGTH. Il valore LENGTH deve essere un multiplo del numero di byte dell'elemento di dati. In caso contrario STATUS = 8088H, ERROR = 1 e non si verifica alcuna trasmissione.
	String	Viene trasferito lo schema di memoria completo del formato della stringa. Il valore LENGTH deve contenere byte per la lunghezza massima e la lunghezza effettiva, e i caratteri della stringa. Tutte le lunghezze e i caratteri del tipo di dati STRING hanno una dimensione di un byte. Se una stringa viene utilizzata come parametro attuale nel parametro DATA, il valore LENGTH deve includere anche due byte per i due campi della lunghezza.

8.6.6 Istruzione RCV_PTP



RCV_PTP (Ricevi punto a punto) controlla se il CM ha ricevuto dei messaggi e, in caso affermativo, li trasferisce dal modulo nella CPU. Un errore restituisce il valore STATUS appropriato.

Il valore STATUS è valido se NDR o ERROR sono veri e specifica il motivo per cui è stata interrotta la ricezione nel CM. Generalmente si tratta di un valore positivo, indicante che il messaggio è stato ricevuto e che la ricezione si è conclusa normalmente. Se il valore STATUS è negativo (il bit più significativo del valore esadecimale è impostato), significa che la ricezione è stata interrotta per un errore di parità, di framing o di overrun.

Ogni modulo CM punto a punto è in grado di bufferizzare fino a un massimo di 1 Kbyte. Può trattarsi di un messaggio molto lungo o di più messaggi brevi.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
EN_R	IN	Bool	Quando questo ingresso è vero controllare se il modulo CM ha ricevuto dei messaggi. Se è stato ricevuto un messaggio, viene trasferito dal modulo nella CPU. Se EN_R è falsa il CM viene controllato per verificare se ha ricevuto messaggi e l'uscita STATUS viene impostata, però il messaggio non viene trasferito nella CPU.
PORT	IN	PORT	Identificativo della porta di comunicazione: questo indirizzo logico è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.
BUFFER	IN	Versione	Questo parametro punta all'indirizzo iniziale del buffer di ricezione. Questo buffer deve essere abbastanza grande da poter ricevere un messaggio con la lunghezza massima. I dati e gli array booleani non sono supportati.
NDR	OUT	Bool	È vero per un ciclo di scansione se i nuovi dati sono pronti e l'operazione si è conclusa senza errori.
ERROR	OUT	Bool	È vero per un ciclo di scansione se l'operazione si è conclusa con un errore
STATUS	OUT	WORD	Codice della condizione di esecuzione
LENGTH	OUT	UInt	Lunghezza del messaggio restituito (in bytes)

STATUS (W#16#...)	Descrizione
0000	Buffer non presente
80E0	Messaggio concluso perché il buffer di ricezione è pieno
80E1	Messaggio concluso per errore di parità
80E2	Messaggio concluso per errore di framing
80E3	Messaggio concluso per errore di overrun

STATUS (W#16#...)	Descrizione
80E4	Messaggio concluso perché la lunghezza calcolata supera le dimensioni del buffer
0094	Messaggio concluso perché è stata ricevuta la lunghezza massima dei caratteri
0095	Messaggio concluso per timeout dei messaggi
0096	Messaggio concluso per timeout intercaratteri
0097	Messaggio concluso per timeout della risposta
0098	Messaggio concluso perché è stata soddisfatta la condizione di lunghezza "N+LEN+M"
0099	Messaggio concluso perché è stata soddisfatta la condizione di fine sequenza

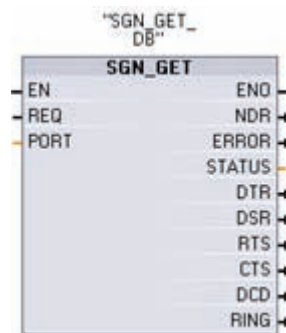
8.6.7 Istruzione RCV_RST



RCV_RST (Resetta buffer di ricezione) cancella il buffer di ricezione nel CM.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Attiva il reset del buffer di ricezione in seguito a un fronte di salita in questo ingresso di abilitazione.
PORT	IN	PORT	Identificativo della porta di comunicazione: la porta deve essere specificata tramite l'indirizzo logico del modulo.
DONE	OUT	Bool	Se è vero per un ciclo di scansione indica che l'ultima richiesta si è conclusa senza errori.
ERROR	OUT	Bool	Se è vero indica che l'ultima richiesta si è conclusa senza errori. Inoltre quando questa uscita è vera, l'uscita STATUS contiene i relativi codici di errore.
STATUS	OUT	WORD	Codice di errore

8.6.8 Istruzione SGN_GET



SGN_GET (Leggi segnali RS232) legge gli stati attuali dei segnali di comunicazione RS232. Questa funzione è valida solo per il CM (modulo di comunicazione) RS232.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Leggi i valori di stato dei segnali RS232 in seguito a un fronte di salita di questo ingresso
PORT	IN	PORT	Identificativo della porta di comunicazione: questo indirizzo logico è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.
NDR	OUT	Bool	È vero per un ciclo di scansione se i nuovi dati sono pronti e l'operazione si è conclusa senza errori.
ERROR	OUT	Bool	È vero per un ciclo di scansione se l'operazione si è conclusa con un errore
STATUS	OUT	WORD	Codice della condizione di esecuzione
DTR	OUT	Bool	Terminale dati pronto, modulo pronto (uscita)
DSR	OUT	Bool	Set di dati pronto, partner della comunicazione pronto (ingresso)
RTS	OUT	Bool	Richiesta di trasmettere, modulo pronto a trasmettere (uscita)
CTS	OUT	Bool	Pronto per la comunicazione, il partner della comunicazione può ricevere i dati (ingresso)
DCD	OUT	Bool	Rileva portante, livello del segnale di ricezione (sempre falso, non supportato)
RING	OUT	Bool	Indicatore di squillo, segnala una chiamata in arrivo (sempre falso, non supportato)

STATUS (W#16#....)	Descrizione
80F0	Il CM è un modulo RS485 e non sono presenti segnali
80F1	Impossibile impostare i segnali a causa del controllo del flusso hardware
80F2	Impossibile impostare DSR perché il modulo è DTE
80F3	Impossibile impostare DTR perché il modulo è DCE

8.6.9 Istruzione SGN_SET



SGN_GET (Imposta segnali RS232) imposta gli stati dei segnali di comunicazione RS232. Questa funzione è valida solo per il CM (modulo di comunicazione) RS232.

Parametro	Tipo di parametro	Tipo di dati	Descrizione
REQ	IN	Bool	Avvia l'impostazione dei segnali RS232 in seguito a un fronte di salita di questo ingresso
PORT	IN	PORT	Identificativo della porta di comunicazione: questo indirizzo logico è una costante che può essere indirizzata nella scheda "Costanti" della tabella delle variabili di default.
SIGNAL	IN	BYTE	Seleziona il segnale da impostare: (anche più di uno) • 01H = imposta RTS • 02H = imposta DTR • 04H = imposta DSR
RTS	IN	Bool	Richiesta di trasmettere, modulo pronto a trasmettere (vero o falso)
DTR	IN	Bool	Terminale dati pronto, modulo pronto a trasmettere il valore da impostare (vero o falso)
DSR	IN	Bool	Set di dati pronto (solo per le interfacce DCE) (non utilizzato)
DONE	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa senza errori
ERROR	OUT	Bool	È vero per un ciclo di scansione se l'ultima richiesta si è conclusa con un errore
STATUS	OUT	WORD	Codice della condizione di esecuzione

STATUS (W#16#....)	Descrizione
80F0	Il CM è un modulo RS485 e non sono presenti segnali impostabili
80F1	Impossibile impostare i segnali a causa del controllo del flusso hardware
80F2	Impossibile impostare DSR perché il modulo è DTE
80F3	Impossibile impostare DTR perché il modulo è DCE

8.7 Errori

Valori restituiti dalle istruzioni PtP

Ogni istruzione PtP ha tre uscite che ne indicano lo stato di completamento.

Parametro	Tipo di dati	Default	Descrizione
DONE	Booleano	Falso	Se è vero per un ciclo di scansione indica che l'ultima richiesta si è conclusa senza errori.
ERROR	Booleano	Falso	Vero indica che l'ultima richiesta si è conclusa con errori, con il codice di errore applicabile in STATUS.
STATUS	WORD	0	Due byte che contengono la classe e il numero dell'errore (se applicabili). STATUS mantiene il proprio valore durante l'esecuzione della funzione.

Classi di errore ed errori comuni

Classe	Classi di errore	Descrizione
Configurazione della porta	80Ax	Definisce errori comuni di configurazione delle porte
Configurazione della trasmissione	80Bx	Definisce errori comuni di configurazione della trasmissione
Configurazione della ricezione	80Cx	Definisce errori comuni di configurazione della ricezione
Runtime di trasmissione	80Dx	Definisce errori comuni di runtime di trasmissione
Runtime di ricezione	80Ex	Definisce errori comuni di runtime di ricezione
Gestione dei segnali	80Fx	Definisce errori comuni di gestione dei segnali

Errori di configurazione delle porte

ID di evento / errore	Descrizione
0x80A0	Il protocollo specificato non esiste
0x80A1	Il baud rate specificato non esiste
0x80A2	La parità specificata non esiste
0x80A3	Il numero di bit di dati specificato non esiste
0x80A4	Il numero di bit di stop specificato non esiste
0x80A5	Il tipo di controllo del flusso specificato non esiste

Errori di configurazione della trasmissione

ID di evento / errore	Descrizione
0x80B0	Il protocollo specificato non esiste
0x80B1	Il baud rate specificato non esiste
0x80B2	La parità specificata non esiste
0x80B3	Il numero di bit di dati specificato non esiste
0x80B4	Il numero di bit di stop specificato non esiste
0x80B5	Il tipo di controllo del flusso specificato non esiste

Errori di configurazione della ricezione

ID di evento / errore	Descrizione
0x80C0	Errore della condizione di avvio
0x80C1	Errore della condizione di fine
0x80C3	Errore di lunghezza massima
0x80C4	Errore di valore N (fare riferimento a N+LEN+M)
0x80C5	Errore di lunghezza (fare riferimento a MAXLEN o a N+LEN+M)
0x80C6	Errore di valore M (fare riferimento a N+LEN+M)
0x80C7	Errore di valore N-lunghezza-M (fare riferimento a N+LEN+M)
0x80C8	Errore di timeout della risposta, non sono stati ricevuti messaggi durante il periodo di ricezione specificato (fare riferimento a RCVTIME o MSGTIME)
0x80C9	Errore di timeout intercaratteri (fare riferimento a CHARGAP)
0x80CA	Errore di timeout per linea inattiva (fare riferimento a Idle Line)
0x80CB	Una sequenza di fine specificata è stata configurata solo con caratteri "don't care"
0x80CC	Una sequenza di inizio specificata è stata configurata solo con caratteri "don't care"

Errori di segnale

ID di evento / errore	Descrizione
0x80F0	Il modulo di comunicazione è un RS485 e non sono presenti segnali
0x80F1	Il modulo di comunicazione è un RS232 ma i segnali non sono impostabili perché è attivo il controllo del flusso hardware
0x80F2	Non è possibile impostare il segnale DSR perché il modulo è un dispositivo DTE

Errori di runtime di trasmissione

ID di evento / errore	Descrizione
Limite del buffer	È stata superata la capacità complessiva del buffer di trasmissione del CP
0x80D0	È stata ricevuta una nuova richiesta mentre il trasmettitore era attivo
0x80D1	Il ricevente ha trasmesso una richiesta di controllo del flusso per sospendere una trasmissione in corso e non ha riattivato la trasmissione durante il tempo di attesa specificato Questo errore viene generato anche durante il controllo del flusso hardware se il ricevente non invia un CTS entro il tempo di attesa specificato
0x80D2	La richiesta di trasmissione è stata interrotta perché non è stato ricevuto alcun segnale DSR dal DCE
0x80D3	È stata superata la capacità complessiva del buffer di trasmissione del CP
0x7000	La funzione di trasmissione non è occupata
0x7001	La funzione di trasmissione è occupata dalla prima chiamata
0x7002	La funzione di trasmissione è occupata con una sequenza di chiamate (interrogazioni dopo la prima chiamata)

Valori di ritorno durante il runtime di ricezione

ID di evento / errore	Descrizione
0x80E0	Il messaggio è stato concluso perché il buffer di ricezione è pieno
0x80E1	Il messaggio è stato concluso a causa di un errore di parità
0x80E2	Il messaggio è stato concluso a causa di un errore di framing
0x80E3	Il messaggio è stato concluso a causa di un errore di overrun
0x80E4	Il messaggio è stato concluso perché la lunghezza specificata supera la dimensione complessiva del buffer
0x0094	Messaggio concluso perché è stata ricevuta la lunghezza massima di caratteri (MAXLEN)
0x0095	Messaggio concluso perché non è stato ricevuto interamente entro il tempo specificato (MSGTIME)
0x0096	Messaggio concluso perché il carattere successivo non è stato ricevuto entro il tempo intercaratteri (CHARGAP)
0x0097	Messaggio concluso perché il primo carattere non è stato ricevuto entro il tempo specificato (RCVTIME)
0x0098	Messaggio concluso perché è stata soddisfatta la condizione per la lunghezza "n+len+m" (N+LEN+M)
0x0099	Messaggio concluso perché è stata soddisfatta la sequenza di fine (ENDSEQ)

Errori vari relativi ai parametri

ID di evento / errore	Descrizione
0x8n3A	È stato indicato un puntatore non ammesso nel parametro n
0x8070	Tutta la memoria di istanza interna è stata utilizzata
0x8080	Numero di porta non valido
0x8082	La parametrizzazione non è stata eseguita perché è già in corso in background
0x8083	Overflow del buffer. Il CM ha restituito più dati del consentito.
0x8085	Il parametro LEN ha valore 0 o è maggiore del valore massimo consentito
0x8088	Il parametro LEN è maggiore dell'area di memoria specificata in DATA

Tool online e di diagnostica

9.1 LED di stato

La CPU e i moduli di I/O si servono di LED per fornire informazioni sullo stato operativo del modulo o degli I/O. LA CPU dispone dei seguenti indicatori di stato:

- STOP/RUN
 - Una luce arancione fissa segnala il modo STOP
 - Una luce verde fissa segnala il modo RUN
 - Una luce lampeggiante (alternativamente verde e arancione) indica che la CPU è in fase di avviamento
- ERROR
 - Una luce rossa lampeggiante indica che si è verificato un errore, ad es. un errore interno alla CPU, un errore con la memory card o un errore di configurazione (i moduli non corrispondono)
 - Una luce rossa fissa indica un guasto hardware
- Quando si inserisce la memory card, il LEDMAINT (manutenzione) lampeggia. La CPU passa quindi in STOP. Dopo che la CPU è passata in STOP, eseguire una delle seguenti funzioni per avviare la valutazione della memory card:
 - Commutare la CPU in RUN
 - Eseguire un reset della memoria (MRES)
 - Riaccendere la CPU.

Descrizione	STOP/RUN Arancione / verde	ERROR Rosso	MAINT Arancione
Alimentazione disinserita	Off	Off	Off
Avviamento, autotest, aggiornamento del firmware	Lampeggiante (alternativamente verde e arancione)	-	Off
Modo STOP	On (arancione)	-	-
Modo RUN	On (arancione)	-	-
Estrarre la memory card	On (arancione)	-	Lampeggiante
Errore	On (arancione o verde)	Lampeggiante	-
Richiesta di manutenzione	On (arancione o verde)	-	On

Descrizione	STOP/RUN Arancione / verde	ERROR Rosso	MAINT Arancione
Guasto hardware	On (arancione)	On	Off
Test dei LED o firmware della CPU difettoso	Lampeggiante (alternativamente verde e arancione)	Lampeggiante	Lampeggiante

La CPU dispone inoltre di due LED che indicano lo stato della comunicazione PROFINET. I LED PROFINET si trovano sotto il coperchio della morsettiera posta in basso.

- Link (verde): se acceso indica che il collegamento è stato stabilito correttamente
- Rx/Tx (giallo): se acceso indica che è in corso una trasmissione

La CPU e i singoli moduli di I/O (SM) dispongono di un LED I/O Channel per ciascun ingresso e uscita digitale. Il LED I/O Channel (verde) si accende o si spegne per indicare lo stato dei singoli ingressi e uscite.

Inoltre ogni SM digitale dispone di un LED DIAG che ne indica lo stato:

- Verde indica che il modulo è operativo
- Rosso indica che il modulo è difettoso o non è operativo

Ogni SM analogico dispone di un LED I/O Channel per ciascun ingresso e uscita analogici.

- Verde indica che il canale è stato configurato ed è attivo
- Rosso indica che è presente una condizione di errore in un ingresso o un'uscita analogica

Inoltre ogni SM analogico dispone di un LED DIAG che ne indica lo stato:

- Verde indica che il modulo è operativo
- Rosso indica che il modulo è difettoso o non è operativo

L'SM rileva la presenza o assenza di alimentazione nel modulo (alimentazione proveniente dal campo, se necessaria).

Descrizione	DIAG (rosso / verde)	I/O Channel (rosso / verde)
L'alimentazione sul lato del campo è disinserita	Rosso lampeggiante	Rosso lampeggiante
Non configurato o aggiornamento in corso	Verde lampeggiante	Off
Modulo configurato senza errori	On (verde)	On (verde)
Condizione di errore	Rosso lampeggiante	-
Errore di I/O (con diagnostica attiva)	-	Rosso lampeggiante
Errore di I/O (con diagnostica disattivata)	-	On (verde)

9.2 Collegamento online e connessione a una CPU

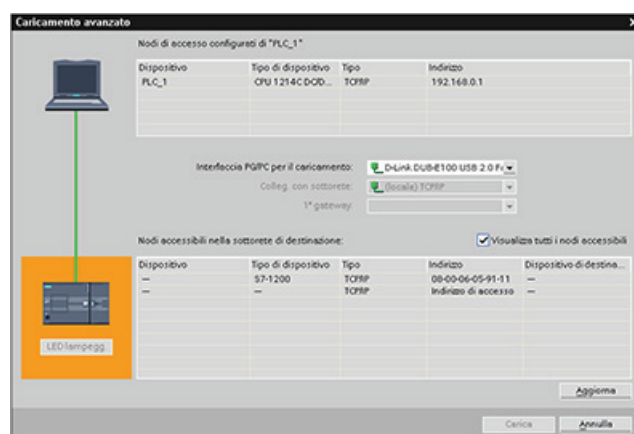
Per poter caricare nel sistema di destinazione i programmi e i dati ingegneristici dei progetti e per eseguire attività come quelle descritte di seguito è necessario stabilire un collegamento online tra il dispositivo di programmazione e il sistema di destinazione:

- Test dei programmi utente
- Visualizzazione e modifica del modo di funzionamento della CPU
- Visualizzazione e impostazione della data e dell'ora della CPU
- Visualizzazione di informazioni sui moduli
- Confronto tra i blocchi online e offline
- Diagnosi dell'hardware

Online &
Diagnostica



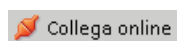
È possibile accedere ai dati del sistema di destinazione nella vista online o di diagnostica tramite la task card "Tool Online".



Lo stato online attuale di un dispositivo è indicato da un'icona visualizzata a destra del dispositivo nella navigazione del progetto.

Il colore arancione indica che è presente un collegamento online.

Selezionare "Nodi accessibili" per cercare una CPU della rete.



Fare clic su "Collega online" per collegarsi a una CPU della rete.

9.3 Impostazione dell'indirizzo IP e dell'ora

È possibile impostare l'indirizzo IP e l'ora nella CPU online.

Dopo aver stabilito il collegamento con una CPU online nell'area "Online e diagnostica" è possibile visualizzare o modificare l'indirizzo IP.

Per maggiori informazioni consultare il paragrafo Indirizzo IP (Pagina 84).

È inoltre possibile visualizzare o impostare i parametri dell'ora e della data della CPU online.



9.4 Pannello operatore CPU per la CPU online

La task card "Pannello operatore CPU" visualizza il modo di funzionamento (STOP o RUN) della CPU online: il pannello indica inoltre se si è verificato un errore nella CPU o se sono presenti valori forzati. È possibile utilizzare un pannello operatore CPU per commutare il modo di funzionamento di una CPU online.

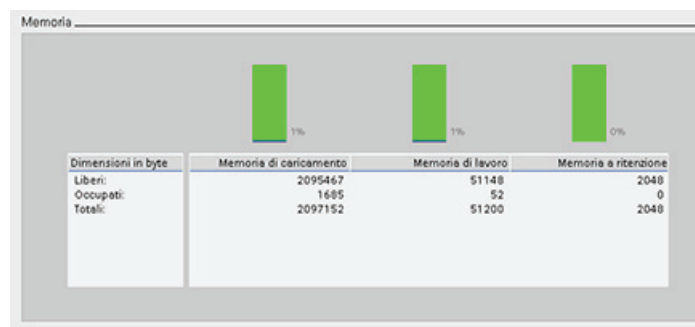


9.5 Controllo del tempo di ciclo e dell'utilizzo della memoria

È possibile controllare il tempo di ciclo e l'utilizzo della memoria nella CPU online.

Dopo aver collegato la CPU online è possibile visualizzare i seguenti dati:

- Tempo di ciclo
- Utilizzo della memoria



9.6 Visualizzazione degli eventi di diagnostica nella CPU

Per prendere visione dell'attività più recente della CPU si utilizza il buffer di diagnostica. Il buffer di diagnostica contiene le seguenti voci:

- Eventi di diagnostica
- Modifiche del modo di funzionamento della CPU (commutazioni da STOP o RUN)

Buffer di diagnostica

Eventi		
✓ Data e ora compresa la differenza oraria CPU/locale		
N°	Data e ora	Evento
1	31/12/1969 19.16.05.679	Stato di funzionamento successivo alla transizione - CPU
2	31/12/1969 19.16.05.677	Richiesta avviata mediante comunicazione: AVVIAMENTO
3	31/12/1969 19.16.05.676	Nuova informazione sul blocco dell'avviamento - stato di
4	31/12/1969 19.15.51.676	Nuova informazione sul blocco dell'avviamento - stato di
5	31/12/1969 19.15.51.476	Nuova informazione sul blocco dell'avviamento - stato di
6	31/12/1969 19.15.40.076	Nuova informazione sul blocco dell'avviamento - stato di
7	31/12/1969 19.00.18.468	Nuova informazione sul blocco dell'avviamento - stato di
8	31/12/1969 19.00.18.462	Stato di funzionamento successivo alla transizione - CPU
9	31/12/1969 19.00.18.203	Rete on - tipo di Memory Card: nessuna Memory Card - CPU

Congela elenco

Dettagli sull'evento: 1 di 9

Informazione CPU: Stato di funzionamento successivo alla transizione
 tipo di avviamento impostato: AVVIAMENTO A CALDO → RUN (se prima della disattivazione su RUN)

blocco dell'avviamento presente:
 - nessun blocco dell'avviamento impostato

CPU passa dallo stato AVVIAMENTO a RUN

Evento entrante

Guida all'evento Apri blocco Salva con nome...

La prima voce contiene l'ultimo evento. Ogni voce del buffer di diagnostica riporta la data e l'ora in cui è stato registrato l'evento e una sua descrizione.

Il numero massimo di voci varia in funzione della CPU e non può essere maggiore di 50.

Vengono memorizzate in modo permanente nel buffer di diagnostica solo le 10 voci più recenti. Se si ripristinano le impostazioni di fabbrica nella CPU il buffer di diagnostica viene resettato e ne vengono cancellate le voci.

9.7 Tabelle per il controllo del programma utente

Le tabelle di controllo consentono di eseguire funzioni di controllo e di comando sui dati man mano che la CPU esegue il programma. I dati possono essere costituiti dall'immagine di processo (I o Q), dagli I/O fisici (I_:P o Q_:P), da M o DB a seconda della funzione di controllo o di comando.

La funzione di controllo non modifica la sequenza del programma ma fornisce informazioni sulla sequenza e i dati del programma nella CPU.

Le funzioni di comando abilitano l'utente al comando della sequenza e dei dati del programma. È importante utilizzarle con cautela perché possono influire sensibilmente sull'esecuzione del programma utente/di sistema. Sono disponibili tre funzioni di comando per la modifica, il forzamento e l'abilitazione delle uscite in STOP.

La tabella di controllo consente di eseguire le seguenti funzioni online:

- Controllo dello stato delle variabili
- Modifica dei valori di singole variabili
- Forzamento di una variabile su un valore specifico

È possibile selezionare quando controllare o modificare la variabile:

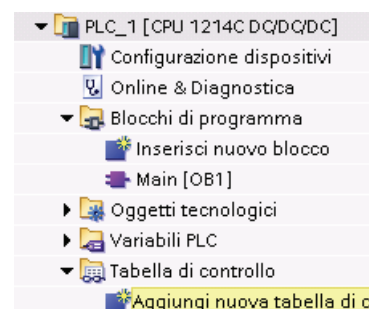
- Inizio ciclo: legge o scrive il valore all'inizio del ciclo di scansione
- Fine ciclo: legge o scrive il valore alla fine del ciclo di scansione
- Commutazione in STOP

Per creare una tabella di controllo:

1. Fare doppio clic su "Aggiungi nuova tabella di controllo" per aprire una nuova tabella di controllo.
2. Immettere il nome della variabile per aggiungere una variabile alla tabella di controllo.

Sono disponibili le seguenti opzioni per il controllo delle variabili:

- Controlla tutto: questo comando avvia il controllo delle variabili visibili nella tabella di controllo attiva.
- Controlla subito: questo comando avvia il controllo delle variabili visibili nella tabella di controllo attiva. La tabella di controllo controlla le variabili immediatamente e una sola volta.



Sono disponibili le seguenti opzioni per la modifica delle variabili:

- "Comanda a 0" imposta a "0" il valore di un indirizzo selezionato.
- "Comanda a 1" imposta a "1" il valore di un indirizzo selezionato.
- "Esegui subito il comando" cambia immediatamente il valore degli indirizzi selezionati per un ciclo di scansione.
- "Comanda con trigger" modifica i valori degli indirizzi selezionati.

Questa funzione non indica in alcun modo che gli indirizzi selezionati sono stati effettivamente modificati. Per avere conferma della modifica utilizzare la funzione "Esegui subito il comando".

- "Abilita uscite di periferia" disattiva il comando di disabilitazione delle uscite ed è disponibile quando la CPU è in STOP.

Per poter controllare le variabili si deve aver stabilito un collegamento online con la CPU.

		Nome	Indirizzo	Formato visu..	Valore di con..	Controlla con trig...	Comanda con trig..
1		"Start"	%I0.0	Bool		Permanente	Permanente
2		"Stop"	%I0.1	Bool		Permanente	Permanente
3		"Running"	%I0.2	Bool		Permanente	Permanente
4		"Del"	%I0.3	Bool		Permanente	Permanente

Le diverse funzioni possono essere selezionate con i pulsanti posti in alto nella tabella di controllo.

Immettere il nome della variabile per controllare e selezionare il formato di visualizzazione nell'elenco a discesa. Se è attivo un collegamento online con la CPU, facendo clic sul pulsante "Controlla" si visualizza il valore effettivo dei dati nel campo "Valore di controllo".

Utilizzo di un trigger durante il controllo o la modifica delle variabili del PLC

La funzione di trigger consente di stabilire in quale punto del ciclo di scansione verrà controllato o modificato l'indirizzo selezionato.

Tipo di trigger	Descrizione
Permanente	Rileva ininterrottamente i dati
Inizio ciclo	Permanente: rileva ininterrottamente i dati all'inizio del ciclo di scansione, dopo che la CPU ha letto gli ingressi
	Una volta: rileva i dati una volta all'inizio del ciclo di scansione, dopo che la CPU ha letto gli ingressi
Fine ciclo	Permanente: rileva ininterrottamente i dati alla fine del ciclo di scansione, prima che la CPU scriva nelle uscite
	Una volta: rileva i dati una volta alla fine del ciclo di scansione, prima che la CPU scriva nelle uscite
Commutazione in STOP	Permanente: rileva ininterrottamente i dati quando la CPU commuta in STOP
	Una volta: rileva i dati una volta dopo che la CPU ha commutato in STOP

Per modificare una variabile PLC in un dato trigger selezionare l'inizio o la fine del ciclo.

- Modifica di un'uscita: il migliore evento di trigger per la modifica di un'uscita è la fine del ciclo di scansione, immediatamente prima che la CPU scriva nelle uscite.

Controllare il valore delle uscite all'inizio del ciclo di scansione per determinare quale valore viene scritto nelle uscite fisiche. Controllare inoltre le uscite prima che la CPU scriva i valori nelle uscite fisiche in modo da verificare la logica del programma e confrontare il comportamento effettivo degli I/O.

- Modifica di un ingresso: il migliore evento di trigger per la modifica di un ingresso è l'inizio del ciclo di scansione, immediatamente dopo che la CPU ha letto gli ingressi e prima che il programma utente ne utilizzi i valori.

Se si modificano gli ingressi all'inizio del ciclo di scansione è necessario controllarli alla fine del ciclo per accertarsi che non abbiano cambiato il loro valore. Se i valori risultano diversi probabilmente il programma utente sta scrivendo erroneamente in un ingresso.

Per capire il motivo per cui la CPU ha commutato in STOP utilizzare il trigger "Commuta in STOP" che rileva gli ultimi valori del processo.

Abilitazione delle uscite in STOP

La tabella di controllo consente di scrivere nelle uscite quando la CPU è in STOP. Questa funzione consente di controllare il cablaggio delle uscite e verificare che il conduttore collegato a un pin di uscita invii un segnale high o low al terminale del dispositivo di processo a cui è collegato.

AVVERTENZA

Anche se la CPU è in STOP l'abilitazione di un'uscita fisica può attivare il punto del processo a cui è collegata.

Se le uscite sono abilitate è possibile modificarne lo stato in STOP. Se sono disabilitate questa possibilità non sussiste.

- Per abilitare la modifica delle uscite in STOP selezionare l'opzione "Abilita uscite di periferia" del comando "Modifica" del menu "Online" oppure fare clic con il tasto destro del mouse sulla riga della tabella di controllo.
- Quando la CPU viene impostata in RUN l'opzione "Abilita uscite di periferia" si disattiva.
- Se alcuni ingressi o uscite sono stati forzati la CPU non è autorizzata ad abilitare le uscite quando è in STOP. Perché questo sia possibile è necessario eliminare la funzione di forzamento.

Forzamento di valori nella CPU

La CPU consente di forzare gli ingressi e le uscite specificandone l'indirizzo fisico (I_:P or Q_:P) nella tabella di controllo e avviando il forzamento.

Nel programma, le letture degli ingressi fisici vengono sovrascritte dal valore forzato. Il programma utilizza il valore forzato per l'elaborazione: quando scrive in un'uscita fisica, ne sovrascrive il valore con il valore forzato. Quest'ultimo diventa disponibile nell'uscita fisica e viene utilizzato dal processo.

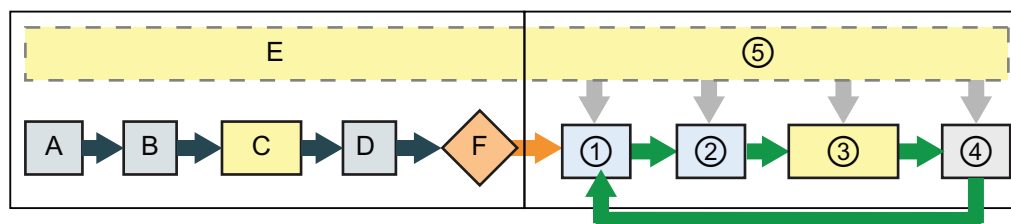
Se si forza un ingresso o un'uscita nella tabella di controllo, le operazioni di forzamento vengono integrate nel programma utente. Anche se il software di programmazione è chiuso, i forzamenti restano attivi nel programma della CPU in funzione finché non vengono resettati dal software di programmazione che imposta la CPU online e arresta la funzione di forzamento. I programmi i cui I/O forzati sono stati caricati in un'altra CPU da una memory card continuano a forzare gli I/O selezionati nel programma.

Se la CPU esegue il programma utente da una memory card con protezione in scrittura, non è possibile avviare o modificare il forzamento degli I/O da una tabella di controllo in quanto la sovrascrittura dei valori nel programma utente protetto in scrittura non è consentita. Ogni tentativo di forzamento dei valori protetti in scrittura genera un errore. Se si utilizza una memory card per il trasferimento di un programma utente, qualsiasi elemento forzato sulla memory card verrà trasferito alla CPU.

Nota

Impossibile forzare gli I/O digitali assegnati a HSC, PWM e PTO

Gli I/O digitali utilizzati dai contatori veloci (HSC), dai dispositivi di modulazione dell'ampiezza degli impulsi (PWM) e di uscita di treni di impulsi (PTO) vengono assegnati durante la configurazione dei dispositivi. Quando gli indirizzi degli I/O digitali vengono assegnati a questi dispositivi, i rispettivi valori non possono essere modificati mediante la funzione di forzamento nella tabella di controllo.



Avviamento

- A Il forzamento non influisce sulla cancellazione dell'area di memoria I.
- B Il forzamento non influisce sull'inizializzazione dei valori di uscita.
- C Durante l'esecuzione degli OB di avviamento la CPU applica il valore di forzamento quando il programma utente accede all'ingresso fisico.
- D Il forzamento non influisce sulla memorizzazione degli eventi di allarme nella coda d'attesa.
- E Il forzamento non influisce sull'abilitazione della scrittura nelle uscite.

RUN

- ① Quando scrive la memoria Q nelle uscite fisiche la CPU applica il valore di forzamento durante l'aggiornamento delle uscite.
- ② Durante la lettura degli ingressi fisici la CPU applica i valori di forzamento subito prima di copiare gli ingressi nella memoria I.
- ③ Durante l'esecuzione del programma utente (OB di ciclo del programma) la CPU applica il valore di forzamento quando il programma utente accede all'ingresso fisico o scrive nell'uscita fisica.
- ④ Il forzamento non influisce sulla gestione delle richieste di comunicazione e della diagnostica di autotest.
- ⑤ Il forzamento non influisce sull'elaborazione degli allarmi in un punto qualsiasi del ciclo di scansione.

Dati tecnici

A.1 Dati tecnici generali

Conformità alle norme

Il sistema di automazione S7-1200 è conforme alle seguenti norme e specifiche per i test. I criteri adottati nei test dell'S7-1200 si basano sulle norme e le specifiche descritte di seguito.

Omologazione CE



Il sistema di automazione S7-1200 soddisfa i requisiti e gli obiettivi di sicurezza stabiliti dalle direttive CE sotto indicate ed è conforme alle norme europee armonizzate (EN) sui controllori a logica programmabile pubblicate nelle Gazzette Ufficiali della Comunità Europea.

- Direttiva EC 2006/95/EC (Direttiva Bassa Tensione) "Materiale elettrico destinato ad essere adoperato entro taluni limiti di tensione"
 - EN 61131-2:2007 Controllori programmabili - Prescrizioni e prove per le apparecchiature
- Direttiva CE 2004/108/EC (Direttiva CEM) "Compatibilità elettromagnetica"
 - Norma sulle emissioni elettromagnetiche
EN 61000-6-4:2007: ambiente industriale
 - Norma sull'immunità elettromagnetica
EN 61000-6-2:2005: ambiente industriale
- Direttiva CE 94/9/EC (ATEX) "Apparecchi e sistemi di protezione destinati ad essere utilizzati in atmosfera potenzialmente esplosiva"
 - EN 60079-15:2005: tipo di protezione 'n'

La Dichiarazione di conformità CE è archiviata e tenuta a disposizione delle autorità competenti presso:

Siemens AG
IA AS RD ST PLC Amberg
Werner-von-Siemens-Str. 50
D92224 Amberg
Germania

Omologazione cULus



Underwriters Laboratories Inc. in conformità a

- Underwriters Laboratories, Inc.: UL 508 Listed (Apparecchiature di controllo per uso industriale)
- Canadian Standards Association: CSA C22.2 n. 142 (Apparecchiature di controllo dei processi)

ATTENZIONE

La serie SIMATIC S7-1200 è conforme alla norma CSA.

Il logo cULus indica che l'S7-1200 è stato verificato e certificato presso gli Underwriters Laboratories (UL) in base alle norme UL 508 e CSA 22.2 n. 142.

Omologazione FM



Factory Mutual Research (FM):

Classe n. 3600 e 3611

Omologato per l'impiego in:

Classe I, Categoria 2, Gruppi di gas A, B, C, D, Classe di temperatura T4A

Ta = 40° C

Classe I, Zona 2, IIC, Classe di temperatura T4 Ta = 40° C

Omologazione ATEX



EN 60079-0:2006: Atmosfere esplosive - Regole generali

EN 60079-15:2005: Costruzioni elettriche per atmosfere esplosive;
tipo di protezione 'n'

II 3 G Ex nA II T4

Rispettare le seguenti condizioni speciali per l'impiego sicuro dell'S7-1200:

- Installare i moduli in un contenitore adeguato garantendo un grado di protezione minimo IP54 secondo quanto stabilito dalla norma EN 60529 e tenendo conto delle condizioni ambientali in cui verranno impiegate le apparecchiature.
- Se in condizioni nominali la temperatura supera i 70° C nel punto di ingresso del cavo o gli 80° C nel punto di diramazione dei conduttori, il cavo in oggetto deve presentare caratteristiche tecniche conformi alla temperatura effettivamente rilevata.
- È necessario adottare misure idonee ad evitare che disturbi transitori non superino di oltre il 40% la tensione nominale.

Omologazione C-Tick



Il sistema di automazione S7-1200 soddisfa i requisiti stabiliti dalle norme AS/NZS 2064 (Classe A)

Omologazione nel settore marittimo

I prodotti S7-1200 vengono periodicamente verificati da enti competenti che ne certificano la conformità alle norme rispetto alle esigenze di particolari settori di mercato e applicazioni. L'elenco aggiornato dei prodotti e delle relative certificazioni può essere richiesto al proprio rappresentante Siemens.

Società di certificazione:

- ABS (American Bureau of Shipping)
- BV (Bureau Veritas)
- DNV (Det Norske Veritas)
- GL (Germanischer Lloyd)
- LRS (Lloyds Register of Shipping)
- Class NK (Nippon Kaiji Kyokai)

Ambienti industriali

Il sistema di automazione S7-1200 è stato progettato per l'utilizzo negli ambienti industriali.

Campo di applicazione	Requisiti relativi all'emissione delle interferenze	Requisiti relativi all'immunità dalle interferenze
Industriale	EN 61000-6-4:2007	EN 61000-6-2:2005

Compatibilità elettromagnetica

La compatibilità elettromagnetica (CEM) è la capacità di un'apparecchiatura elettrica di funzionare nel modo previsto in presenza di interferenze elettromagnetiche e senza generare disturbi elettromagnetici di livello tale da compromettere il funzionamento di altre apparecchiature poste nelle vicinanze.

Compatibilità elettromagnetica - Immunità secondo EN 61000-6-2	
EN 61000-4-2 Scarica elettrostatica	Scarica elettrostatica in aria a 8 kV su tutte le superfici, scarica elettrostatica a contatto a 6 kV sulle superfici conduttive esposte
EN 61000-4-3 Immunità ai campi elettromagnetici irradiati	80 ... 1000 MHz, 10 V/m, 80% AM a 1 kHz 1-4 ... 2,0 GHz, 3 V/m, 80% AM a 1 kHz 2,0 ... 2,7 GHz, 1 V/m, 80% AM a 1 kHz
EN 61000-4-4 Burst transitori veloci	2 kV, 5 kHz con rete di accoppiamento all'alimentazione AC e DC del sistema 2 kV, 5 kHz con accoppiamento agli I/O

Compatibilità elettromagnetica - Immunità secondo EN 61000-6-2	
EN 6100-4-5 Immunità alle sovratensioni	Sistemi AC - modo comune 2 kV, modo differenziale 1kV, sistemi DC - modo comune 2 kV, modo differenziale 1kV. Per i sistemi DC (segnali di I/O, sistemi di alimentazione DC) è richiesta una protezione esterna.
EN 61000-4-6 Disturbi elettromagnetici condotti	150 kHz ... 80 MHz, 10 V RMS, 80% AM a 1kHz
EN 61000-4-11 Buchi di tensione	Sistemi AC 0% per 1 ciclo, 40% per 12 cicli e 70% per 30 cicli a 60 Hz

Compatibilità elettromagnetica - Emissioni condotte e irradiate secondo EN 61000-6-4	
Emissioni condotte EN 55011, Classe A, Gruppo 1 0,15 MHz ... 0,5 MHz 0,5 MHz ... 5 MHz 5 MHz ... 30 MHz	<79dB (µV) quasi picco; <66 dB (µV) media <73dB (µV) quasi picco; <60 dB (µV) media <73dB (µV) quasi picco; <60 dB (µV) media
Emissioni irradiate EN 55011, Classe A, Gruppo 1 30 MHz ... 230 MHz 230 MHz ... 1 GHz	<40dB (µV/m) quasi picco; misurate a 10 m <47dB (µV/m) quasi picco; misurate a 10 m

Condizioni ambientali

Condizioni ambientali Trasporto e immagazzinaggio	
EN 60068-2-2, test Bb, caldo secco e EN 60068-2-1, test Ab, freddo	-40° C ... +70° C
EN 60068-2-30, test Dd, caldo umido	25° C ... 55° C, 95% umidità
EN 60068-2-14, test Na, brusca variazione termica	-40° C ... +70° C, tempo di sosta di 3 ore, 2 cicli
EN 60068-2-32, caduta libera	0,3 m, 5 volte, imballato per la spedizione
Pressione atmosferica	1080 a 660h Pa (corrispondente a un'altitudine di -1000 ... 3500 m)

Condizioni ambientali Esercizio	
Campo di temperatura ambiente (presa d'aria di 25 mm sotto l'unità)	0° C ... 55° C in caso di montaggio orizzontale, 0° C ... 45° C in caso di montaggio verticale 95% di umidità senza condensa
Pressione atmosferica	Da 1080 a 795 hPa (corrispondente a un'altitudine da -1000 a 2000m)
Concentrazione di sostanze inquinanti	SO ₂ : < 0,5 ppm; H ₂ S: < 0,1 ppm; RH < 60% senza condensa
EN 60068-2-14, test Nb, variazione termica	Da 5° C a 55° C, 3° C/minuto
EN 60068-2-27 Sollecitazioni meccaniche	15 G, impulso di 11 ms, 6 urti in ognuno dei 3 assi
EN 60068-2-6 Vibrazione sinusoidale	Montaggio su guida DIN: 3.5 mm da 5 a 9 Hz, 1G da 9 a 150 Hz, montaggio su pannello: 7.0 mm da 5 a -9 Hz; 2 G da 9 a 150 Hz 10 oscillazioni per ogni asse, 1 ottava al minuto

Test di isolamento per alti potenziali	
Circuiti con tensioni nominali a 24 V/5 V	520 V DC (test di tipo dei limiti di isolamento ottico)
Tra i circuiti a 115/230 V e la terra	1,500 V AC test di routine/1950 V DC test di tipo
Tra i circuiti a 115/230 V e i circuiti a 115/230 V	1,500 V AC test di routine/1950 V DC test di tipo
Tra i circuiti a 115 V/230V e i circuiti a 24 V/5 V	1,500 V AC test di routine/3250 V DC test di tipo

Classe di sicurezza

- Classe di sicurezza I secondo EN 61131-2 (il conduttore di protezione non è necessario)

Grado di protezione

- Protezione meccanica IP20, EN 60529
- Protegge dal contatto con alta tensione, come sperimentato su provino standard. Si richiede protezione esterna da polvere, sporcizia, acqua e corpi estranei di diametro < 12,5 mm.

Tensioni nominali

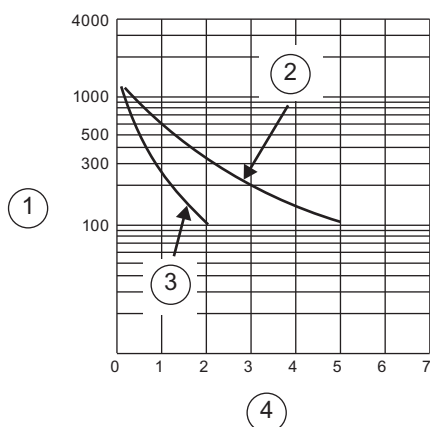
Tensione nominale	Tolleranza
24 V DC	20,4 V DC ... 28,8 V DC
120/230 V AC	85 V AC ... 264 V AC, 47 ... 63 Hz

ATTENZIONE

Quando un contatto meccanico attiva la corrente in uscita verso la CPU S71200 o un'unità di I/O digitale, invia un "1" alle uscite digitali per circa 50 microsecondi. È indispensabile tenerne conto, in particolare se si utilizzano dispositivi che reagiscono a impulsi di breve durata.

Durata di servizio dei relè

La figura più sotto riporta i dati relativi alle prestazioni dei relè forniti dai produttori. Le prestazioni effettive possono variare in base all'applicazione specifica del relè. Per aumentare la durata dei contatti inserire un circuito di protezione esterno adatto al carico.



- ① Durata di servizio (10³ operazioni)
- ② Carico resistivo da 250 V AC
Carico resistivo da 30 V DC
- ③ Carico induttivo da 250 V AC (f. p.=0,4)
Carico induttivo da 30 V DC (L/R=7ms)
- ④ Corrente di esercizio nominale (A)

A.2 CPU

A.2.1 Dati tecnici della CPU 1211C

Dati tecnici			
Modello	CPU 1211C AC/DC/relè	CPU 1211C DC/DC/relè	CPU 1211C DC/DC/DC
Numero di ordinazione	6ES7 211-1BD30-0XB0	6ES7 211-1HD30-0XB0	6ES7 211-1AD30-0XB0
Dati generali			
Dimensioni L x A x P (mm)	90 x 100 x 75		
Peso	420 g	380 g	370 g
Dissipazione di potenza	10 W	8 W	
Corrente disponibile (bus CM)	750 mA max. (5 V DC)		
Corrente disponibile (24 V DC)	300 mA max. (alimentazione sensori)		
Assorbimento di corrente ingressi digitali (24 V DC)	4 mA/ingresso utilizzato		
Caratteristiche della CPU			
Memoria utente	memoria di lavoro 25 Kbyte / memoria di caricamento 1 Mbyte / memoria a ritenzione 2 Kbyte		
I/O digitali on-board	6 ingressi/4 uscite		
I/O analogici on-board	2 ingressi		
Dimensione dell'immagine di processo	1024 byte di ingressi (I)/1024 byte di uscite (Q)		

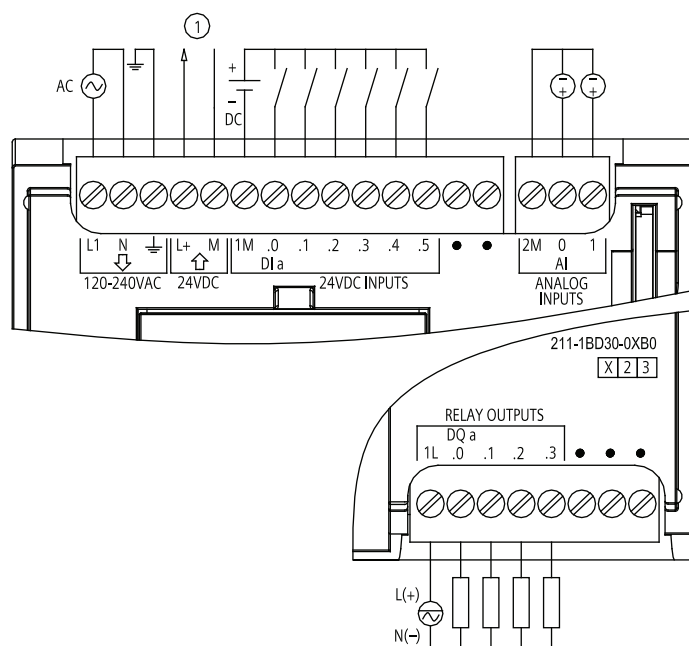
Dati tecnici			
Modello	CPU 1211C AC/DC/relè	CPU 1211C DC/DC/relè	CPU 1211C DC/DC/DC
Memoria di merker (M)	4096 byte		
Ampliamento con moduli di I/O	Nessuno		
Ampliamento con signal board	1 SB max.		
Ampliamento con moduli di comunicazione	3 CM max.		
Contatori veloci	3 in totale A una fase: 3 a 100 kHz in quadratura di fase: 3 a 80 kHz		
Uscite di impulsi	2		
Ingressi di misurazione impulsi	6		
Allarmi di ritardo e di schedulazione orologio	4 in totale con risoluzione di 1 ms		
Allarmi di fronte	6 di fronte di salita e 6 di fronte di discesa (10 e 10 con signal board opzionale)		
Memory card	SIMATIC Memory Card (opzionale)		
Precisione orologio hardware	+/- 60 secondi/mese		
Tempo di ritenzione dell'orologio hardware	Tip. 10 giorni/min. 6 giorni a 40°C (condensatore ad elevata capacità che non richiede manutenzione)		
Prestazioni			
Velocità di esecuzione operazioni booleane	0,1 µs/istruzione		
Velocità di esecuzione di Trasferisci parola	12 µs/istruzione		
Velocità di esecuzione operazioni matematiche con numeri reali	18 µs/istruzione		
Comunicazione			
Numero di porte	1		
Tipo	Ethernet		
Collegamenti	• 3 per gli HMI • 1 per il dispositivo di programmazione • 8 per le istruzioni Ethernet del prorgamma utente • 3 per CPU-CPU		
Velocità dati	10/100 Mb/s		
Isolamento (tra il segnale esterno e la logica PLC)	Isolato da trasformatore, 1500 V DC		
Tipo di cavo	CAT5e schermato		
Alimentazione			
Campo di tensione	85 ... 264 V AC	20,4 ... 28,8 V DC	
Frequenza della linea	47 ... 63 Hz	--	
Corrente in ingresso Solo CPU a pieno carico	60 mA a 120 V AC 30 mA a 240 V AC	300 mA a 24 V DC	
CPU con tutti i moduli di ampliamento a pieno carico	180 mA a 120 V AC 90 mA a 240 V AC	900 mA a 24 V DC	

Dati tecnici			
Modello	CPU 1211C AC/DC/relè	CPU 1211C DC/DC/relè	CPU 1211C DC/DC/DC
Spunto di corrente all'accensione (max.)	20 A a 264 V AC	12 A a 28,8 V DC	
Isolamento (tra l'alimentazione in ingresso e il circuito logico)	1500 V AC	Nessuno	
Dispersione verso terra, dalla linea AC alla terra funzionale	0.5 mA max.	-	
Tempo di mantenimento (in caso di mancanza di alimentazione)	20 ms a 120 V AC 80 ms a 240 V AC	10 ms a 24 V DC	
Fusibile interno, non sostituibile dall'utente	3 A, 250 V, lento		
Alimentazione dei sensori			
Campo di tensione	20,4 ... 28,8 V DC	L+ meno 4 V DC min.	
Corrente di uscita (max.)	300 mA (protetta da cortocircuito)		
Rumore di ondulazione max. (<10 MHz)	< 1 V da picco a picco	Come la linea di ingresso	
Isolamento (tra il circuito logico della CPU e l'alimentazione dei sensori)	Nessuno		
Ingressi digitali			
Numero di ingressi	6		
Tipo	Ad assorbimento/emissione di corrente (secondo IEC "tipo 1" se ad assorbimento di corrente)		
Tensione nominale	24 V DC a 4 mA, nominale		
Tensione continua ammessa	30 V DC max.		
Sovratensione transitoria	35 V DC per 0,5 sec.		
Segnale logico 1 (min.)	15 V DC a 2,5 mA		
Segnale logico 0 (max.)	5 V DC a 1 mA		
Isolamento (tra il campo e i circuiti logici)	500 V AC per 1 minuto		
Gruppi di isolamento	1		
Tempi di filtraggio	0,2; 0,4; 0,8; 1,6; 3,2; 6,4 e 12,8 ms (selezionabili in gruppi di 4)		
Frequenze di clock in ingresso agli HSC (max.) (segnale logico 1 = 15 ... 26 V DC)	A una fase: 100 KHz In quadratura di fase: 80 KHz		
Numero di ingressi ON contemporaneamente	6		
Lunghezza del cavo (metri)	500 schermato, 300 non schermato, 50 schermato per gli ingressi HSC		
Ingressi analogici			
Numero di ingressi	2		
Tipo	Tensione (asimmetrico)		
Campo	0 ... 10 V		
Campo di fondo scala (parola di dati)	0 ... 27648 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		
Campo di overshoot (parola di dati)	27,649 ... 32,511 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		

Dati tecnici			
Modello	CPU 1211C AC/DC/relè	CPU 1211C DC/DC/relè	CPU 1211C DC/DC/DC
Overflow (parola di dati)	32,512 ... 32767 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		
Risoluzione	10 bit		
Tensione di resistenza max.	35 V DC		
Livellamento	Nessuno, Debole, Medio o Forte (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere i tempi di risposta al gradino)		
Filtraggio del rumore	10, 50 o 60 Hz (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere le velocità di campionamento)		
Impedenza	≥100 KΩ		
Isolamento (tra il campo e i circuiti logici)	Nessuno		
Precisione (25°C / 0 ... 55°C)	3,0% / 3,5% del valore di fondo scala		
Reiezione in modo comune	40 dB, DC a 60 Hz		
Campo operativo del segnale	La tensione di segnale più quella di modo comune deve essere inferiore a +12 V e maggiore di -12 V		
Lunghezza del cavo (metri)	100 m, cavo doppio ritorto schermato		
Uscite digitali			
Numero di uscite	4		
Tipo	Relè, contatto Dry		MOSFET a stato solido
Campo di tensione	5 ... 30 V DC oppure 5 ... 250 V AC		20,4 ... 28,8 V DC
Segnale logico 1 a corrente max.	--		20 V DC min.
Segnale logico 0 con carico di 10 KΩ	--		0,1 V DC max.
Corrente (max.)	2,0 A		0,5 A
Carico delle lampade	30 W DC / 200 W AC		5 W
Resistenza in stato ON	0,2 Ω max. da nuova		0,6 Ω max.
Corrente di dispersione per punto	--		10 µA max.
Corrente di spunto	7 A con contatti chiusi		8 A per 100 ms max.
Protezione da sovraccarico	No		
Isolamento (tra il campo e i circuiti logici)	1500 V AC per 1 minuto (tra la bobina e il contatto) Nessuno (tra la bobina e i circuiti logici)		500 V AC per 1 minuto
Resistenza di isolamento	100 MΩ min. da nuova		--
Isolamento tra contatti aperti	750 V AC per 1 minuto		--
Gruppi di isolamento	1		1
Clamp per tensioni induttive	--		L+ meno 48 V DC, dissipazione di 1 W
Ritardo durante la commutazione (Qa.0 ... Qa.3)	10 ms max.		1,0 µs max., da off a on 3,0 µs max., da on a off
Frequenza di uscita treni di impulsi (Qa.0 e Qa.2)	Sconsigliato		100 KHz max., 2 Hz min.
Tempo di vita in cicli meccanici (senza carico)	10,000,000 cicli di apertura/chiusura		--
Durata contatti con carico nominale	100,000 cicli di apertura/chiusura		--

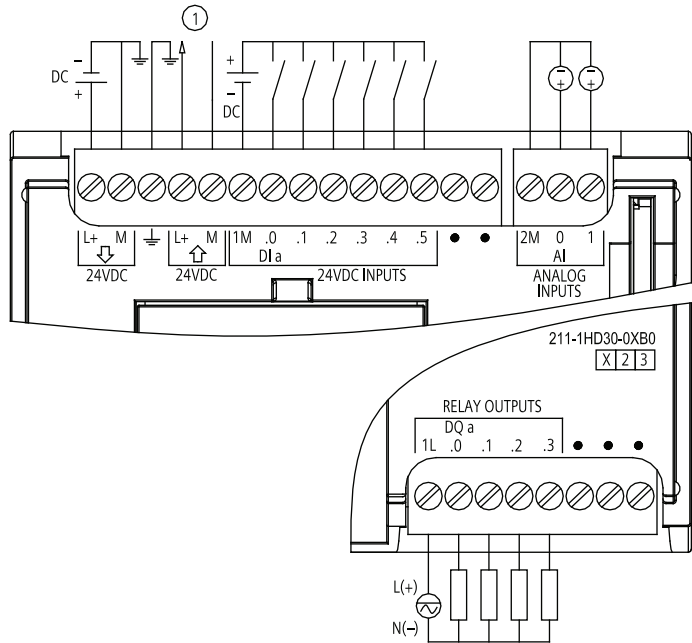
Dati tecnici			
Modello	CPU 1211C AC/DC/relè	CPU 1211C DC/DC/relè	CPU 1211C DC/DC/DC
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)		
Numero di uscite ON contemporaneamente	4		
Lunghezza del cavo (metri)	500 schermato, 150 non schermato		

Schemi elettrici



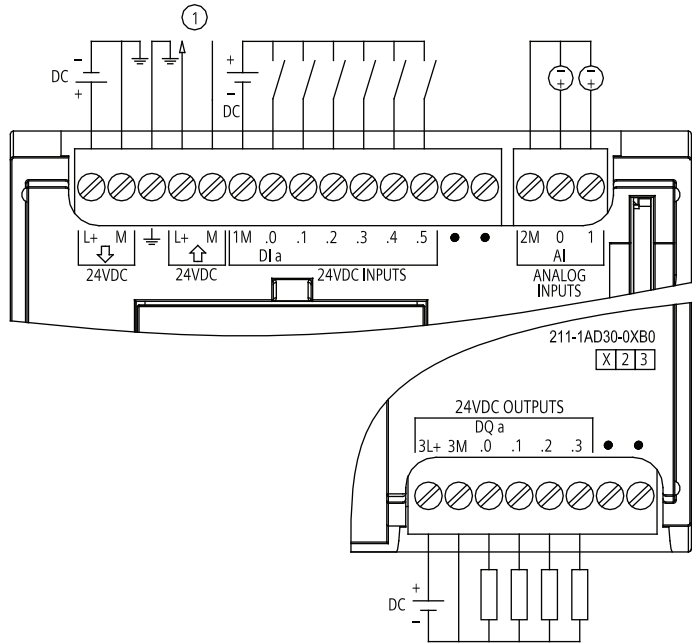
① Uscita di alimentazione per sensori a 24 V DC

Figura A-1 CPU 1211C AC/DC/relè (6ES7 211-1BD30-0XB0)



① Uscita di alimentazione per sensori a 24 V DC

Figura A-2 CPU 1211C DC/DC/relè (6ES7 211-1HD30-0XB0)



① Uscita di alimentazione per sensori a 24 V DC

Figura A-3 CPU 1211C DC/DC/DC (6ES7 211-1AD30-0XB0)

A.2.2 Dati tecnici della CPU 1212C

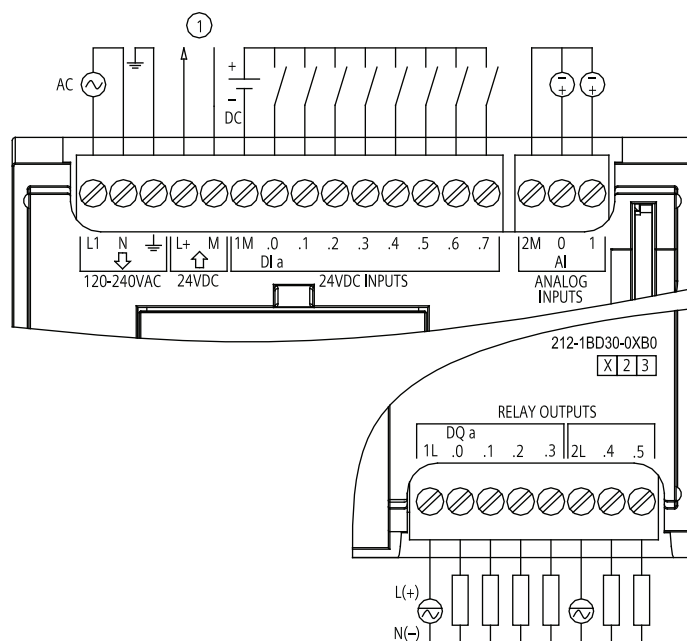
Dati tecnici			
Modello	CPU 1212C AC/DC/relè	CPU 1212C DC/DC/relè	CPU 1212C DC/DC/DC
Numero di ordinazione	6ES7 212-1BD30-0XB0	6ES7 212-1HD30-0XB0	6ES7 212-1AD30-0XB0
Dati generali			
Dimensioni L x A x P (mm)	90 x 100 x 75		
Peso	425 g	385 g	370 g
Dissipazione di potenza	11 W	9 W	
Corrente disponibile (SM e bus CM)	1000 mA max. (5 V DC)		
Corrente disponibile (24 V DC)	300 mA max. (alimentazione sensori)		
Assorbimento di corrente ingressi digitali (24 V DC)	4 mA/ingresso utilizzato		
Caratteristiche della CPU			
Memoria utente	memoria di lavoro 25 Kbyte / memoria di caricamento 1 Mbyte / memoria a ritenzione 2 Kbyte		
I/O digitali on-board	8 ingressi/6 uscite		
I/O analogici on-board	2 ingressi		
Dimensione dell'immagine di processo	1024 byte di ingressi (I)/1024 byte di uscite (Q)		
Memoria di merker (M)	4096 byte		
Ampliamento con moduli di I/O	2 SM max.		
Ampliamento con signal board	1 SB max.		
Ampliamento con moduli di comunicazione	3 CM max.		
Contatori veloci	4 in totale A una fase: 3 a 100 kHz e 1 con frequenza di clock di 30 kHz In quadratura di fase: 3 a 80 kHz e 1 con frequenza di clock di 20 kHz		
Uscite di impulsi	2		
Ingressi di misurazione impulsi	8		
Allarmi di ritardo e di schedulazione orologio	4 in totale con risoluzione di 1 ms		
Allarmi di fronte	8 di fronte di salita e 8 di fronte di discesa (12 e 12 con signal board opzionale)		
Memory card	SIMATIC Memory Card (opzionale)		
Precisione orologio hardware	+/- 60 secondi/mese		
Tempo di ritenzione dell'orologio hardware	Tip. 10 giorni/min. 6 giorni a 40°C (condensatore ad elevata capacità che non richiede manutenzione)		
Prestazioni			
Velocità di esecuzione operazioni booleane	0,1 µs/istruzione		
Velocità di esecuzione di Trasferisci parola	12 µs/istruzione		
Velocità di esecuzione operazioni matematiche con numeri reali	18 µs/istruzione		

Dati tecnici			
Modello	CPU 1212C AC/DC/relè	CPU 1212C DC/DC/relè	CPU 1212C DC/DC/DC
Comunicazione			
Numero di porte	1		
Tipo	Ethernet		
Collegamenti	• 3 per gli HMI • 1 per il dispositivo di programmazione • 8 per le istruzioni Ethernet del prorgamma utente • 3 per CPU-CPU		
Velocità dati	10/100 Mb/s		
Isolamento (tra il segnale esterno e la logica PLC)	Isolato da trasformatore, 1500 V DC		
Tipo di cavo	CAT5e schermato		
Alimentazione			
Campo di tensione	85 ... 264 V AC	20,4 ... 28,8 V DC	
Frequenza della linea	47 ... 63 Hz	--	
Corrente in ingresso Solo CPU a pieno carico	80 mA a 120 V AC 40 mA a 240 V AC	400 mA at 24 V DC	
CPU con tutti i moduli di ampliamento a pieno carico	240 mA a 120 V AC 120 mA a 240 V AC	1200 mA a 24 V DC	
Spunto di corrente all'accensione (max.)	20 A a 264 V AC	12 A a 28,8 V DC	
Isolamento (tra l'alimentazione in ingresso e il circuito logico)	1500 V AC	Nessuno	
Dispersione verso terra, dalla linea AC alla terra funzionale	0.5 mA max.	-	
Tempo di mantenimento (in caso di mancanza di alimentazione)	20 ms a 120 V AC 80 ms a 240 V AC	10 ms a 24 V DC	
Fusibile interno, non sostituibile dall'utente	3 A, 250 V, lento		
Alimentazione dei sensori			
Campo di tensione	20,4 ... 28,8 V DC	L+ meno 4 V DC min.	
Corrente di uscita (max.)	300 mA (protetta da cortocircuito)		
Rumore di ondulazione max. (<10 MHz)	< 1 V da picco a picco	Come la linea di ingresso	
Isolamento (tra il circuito logico della CPU e l'alimentazione dei sensori)	Nessuno		
Ingressi digitali			
Numero di ingressi	8		
Tipo	Ad assorbimento/emissione di corrente (secondo IEC "tipo 1" se ad assorbimento di corrente)		
Tensione nominale	24 V DC a 4 mA, nominale		
Tensione continua ammessa	30 V DC max.		
Sovratensione transitoria	35 V DC per 0,5 sec.		
Segnale logico 1 (min.)	15 V DC a 2,5 mA		

Dati tecnici			
Modello	CPU 1212C AC/DC/relè	CPU 1212C DC/DC/relè	CPU 1212C DC/DC/DC
Segnale logico 0 (max.)	5 V DC a 1 mA		
Isolamento (tra il campo e i circuiti logici)	500 V AC per 1 minuto		
Gruppi di isolamento	1		
Tempi di filtraggio	0,2; 0,4; 0,8; 1,6; 3,2; 6,4 e 12,8 ms (selezionabili in gruppi di 4)		
Frequenze di clock in ingresso agli HSC (max.) (segnale logico 1 = 15 ... 26 V DC)	A una fase: 100 KHz (Ia.0 ... Ia.5) e 30 KHz (Ia.6 ... Ia.7) In quadratura di fase: 80 KHz (Ia.0 ... Ia.5) e 20 KHz (Ia.6 ... Ia.7)		
Numero di ingressi ON contemporaneamente	8		
Lunghezza del cavo (metri)	500 schermato, 300 non schermato, 50 schermato per gli ingressi HSC		
Ingressi analogici			
Numero di ingressi	2		
Tipo	Tensione (asimmetrico)		
Campo	0 ... 10 V		
Campo di fondo scala (parola di dati)	0 ... 27648 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		
Campo di overshoot (parola di dati)	27,649 ... 32,511 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		
Overflow (parola di dati)	32,512 ... 32767 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		
Risoluzione	10 bit		
Tensione di resistenza max.	35 V DC		
Livellamento	Nessuno, Debole, Medio o Forte (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere i tempi di risposta al gradino)		
Filtraggio del rumore	10, 50 o 60 Hz (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere le velocità di campionamento)		
Impedenza	≥100 KΩ		
Isolamento (tra il campo e i circuiti logici)	Nessuno		
Precisione (25°C / 0 ... 55°C)	3,0% / 3,5% del valore di fondo scala		
Reiezione in modo comune	40 dB, DC a 60 Hz		
Campo operativo del segnale	La tensione di segnale più quella di modo comune deve essere inferiore a +12 V e maggiore di -12 V		
Lunghezza del cavo (metri)	100 cavo doppio ritorto schermato		
Uscite digitali			
Numero di uscite	6		
Tipo	Relè, contatto Dry		MOSFET a stato solido
Campo di tensione	5 ... 30 V DC oppure 5 ... 250 V AC		20,4 ... 28,8 V DC
Segnale logico 1 a corrente max.	--		20 V DC min.
Segnale logico 0 con carico di 10 KΩ	--		0,1 V DC max.
Corrente (max.)	2.0 A		0,5 A
Carico delle lampade	30 W DC / 200 W AC		5 W

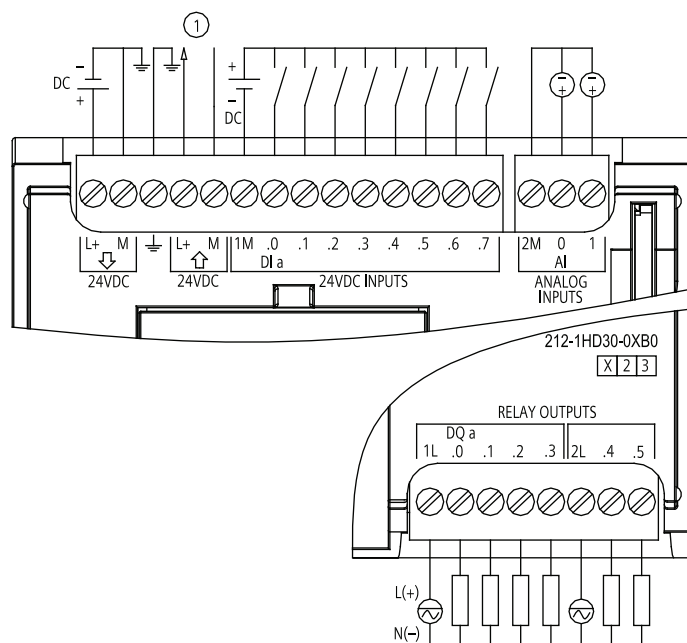
Dati tecnici			
Modello	CPU 1212C AC/DC/relè	CPU 1212C DC/DC/relè	CPU 1212C DC/DC/DC
Resistenza in stato ON	0,2 Ω max. da nuova		0,6 Ω max.
Corrente di dispersione per punto	--		10 μ A max.
Corrente di spunto	7 A con contatti chiusi		8 A per 100 ms max.
Protezione da sovraccarico	No		
Isolamento (tra il campo e i circuiti logici)	1500 V AC per 1 minuto (tra la bobina e il contatto) Nessuno (tra la bobina e i circuiti logici)		500 V AC per 1 minuto
Resistenza di isolamento	100 M Ω min. da nuova		--
Isolamento tra contatti aperti	750 V AC per 1 minuto		--
Gruppi di isolamento	2		1
Clamp per tensioni induttive	--		L+ meno 48 V DC, dissipazione di 1 W
Ritardo durante la commutazione (Qa.0 ... Qa.3)	10 ms max.		1,0 μ s max., da off a on 3,0 μ s max., da on a off
Ritardo durante la commutazione (Qa.4 ... Qa.5)	10 ms max.		50 μ s max., da off a on 200 μ s max., da on a off
Frequenza di uscita treni di impulsi (Qa.0 e Qa.2)	Sconsigliato		100 KHz max., 2 Hz min.
Tempo di vita in cicli meccanici (senza carico)	10,000,000 cicli di apertura/chiusura		--
Durata contatti con carico nominale	100,000 cicli di apertura/chiusura		--
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)		
Numero di uscite ON contemporaneamente	6		
Lunghezza del cavo (metri)	500 schermato, 150 non schermato		

Schemi elettrici



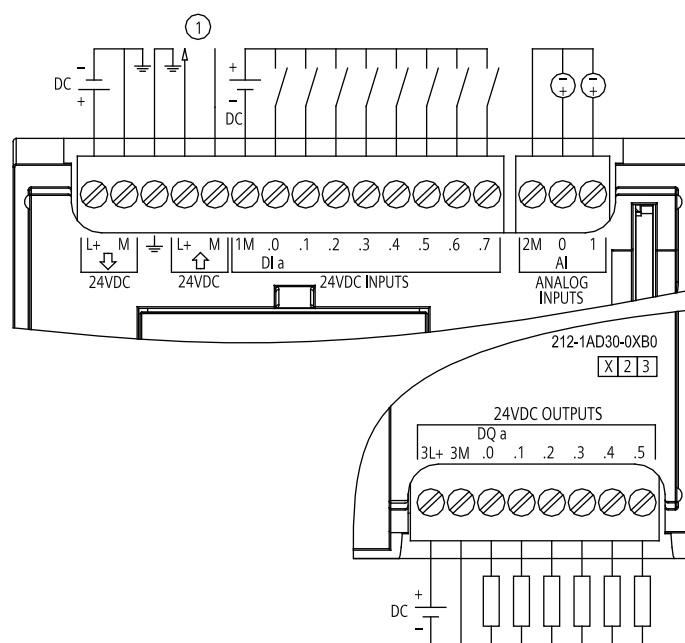
① Uscita di alimentazione per sensori a 24 V DC

Figura A-4 CPU 1212C AC/DC relè (6ES7 212-1BD30-0XB0)



① Uscita di alimentazione per sensori a 24 V DC

Figura A-5 CPU 1212C DC/DC/relè (6ES7 212-1HD30-0XB0)



① Uscita di alimentazione per sensori a 24 V DC

Figura A-6 CPU 1212C DC/DC/DC (6ES7 212-1AD30-0XB0)

A.2.3 Dati tecnici della CPU 1214C

Dati tecnici			
Modello	CPU 1214C AC/DC/relè	CPU 1214C DC/DC/relè	CPU 1214C DC/DC/DC
Numero di ordinazione	6ES7 214-1BE30-0XB0	6ES7 214-1HE30-0XB0	6ES7 214-1AE30-0XB0
Dati generali			
Dimensioni L x A x P (mm)	110 x 100 x 75		
Peso	475 g	435 g	415 g
Dissipazione di potenza	14 W	12 W	
Corrente disponibile (SM e bus CM)	1600 mA max. (5 V DC)		
Corrente disponibile (24 V DC)	400 mA max. (alimentazione sensori)		
Assorbimento di corrente ingressi digitali (24 V DC)	4 mA/ingresso utilizzato		
Caratteristiche della CPU			
Memoria utente	memoria di lavoro 50 Kbyte / memoria di caricamento 2 Mbyte / memoria a ritenzione 2 Kbyte		
I/O digitali on-board	14 ingressi/10 uscite		
I/O analogici on-board	2 ingressi		
Dimensione dell'immagine di processo	1024 byte di ingressi (I)/1024 byte di uscite (Q)		
Memoria di merker (M)	8192 byte		
Ampliamento con moduli di I/O	8 SM max.		

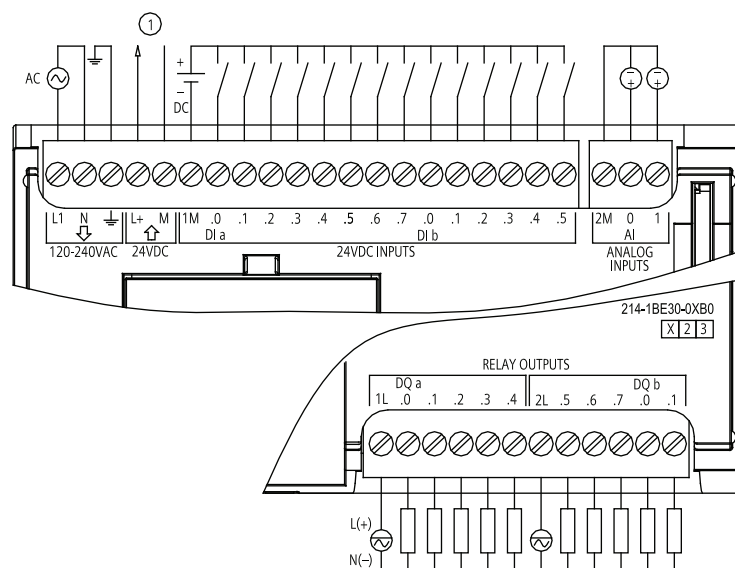
Dati tecnici			
Modello	CPU 1214C AC/DC/relè	CPU 1214C DC/DC/relè	CPU 1214C DC/DC/DC
Ampliamento con signal board	1 SB max.		
Ampliamento con moduli di comunicazione	3 CM max.		
Contatori veloci	6 in totale A una fase: 3 a 100 kHz e 3 con frequenza di clock di 30 kHz In quadratura di fase: 3 a 80 kHz e 3 con frequenza di clock di 20 kHz		
Uscite di impulsi	2		
Ingressi di misurazione impulsi	14		
Allarmi di ritardo e di schedulazione orologio	4 in totale con risoluzione di 1 ms		
Allarmi di fronte	12 di fronte di salita e 12 di fronte di discesa (14 e 14 con signal board opzionale)		
Memory card	SIMATIC Memory Card (opzionale)		
Precisione orologio hardware	+/- 60 secondi/mese		
Tempo di ritenzione dell'orologio hardware	Tip. 10 giorni/min. 6 giorni a 40°C (condensatore ad elevata capacità che non richiede manutenzione)		
Prestazioni			
Velocità di esecuzione operazioni booleane	0,1 µs/istruzione		
Velocità di esecuzione di Trasferisci parola	12 µs/istruzione		
Velocità di esecuzione operazioni matematiche con numeri reali	18 µs/istruzione		
Comunicazione			
Numero di porte	1		
Tipo	Ethernet		
Collegamenti	• 3 per gli HMI • 1 per il dispositivo di programmazione • 8 per le istruzioni Ethernet del prorgamma utente • 3 per CPU-CPU		
Velocità dati	10/100 Mb/s		
Isolamento (tra il segnale esterno e la logica PLC)	Isolato da trasformatore, 1500 V DC		
Tipo di cavo	CAT5e schermato		
Alimentazione			
Campo di tensione	85 ... 264 V AC	20,4 ... 28,8 V DC	
Frequenza della linea	47 ... 63 Hz	--	
Corrente in ingresso Solo CPU a pieno carico	100 mA a 120 V AC 50 mA a 240 V AC	500 mA a 24 V DC	
CPU con tutti i moduli di ampliamento a pieno carico	300 mA a 120 V AC 150 mA a 240 V AC	1500 mA a 24 V DC	
Spunto di corrente all'accensione (max.)	20 A a 264 V AC	12 A a 28,8 V DC	

Dati tecnici			
Modello	CPU 1214C AC/DC/relè	CPU 1214C DC/DC/relè	CPU 1214C DC/DC/DC
Isolamento (tra l'alimentazione in ingresso e il circuito logico)	1500 V AC	Nessuno	
Dispersione verso terra, dalla linea AC alla terra funzionale	0.5 mA max.	-	
Tempo di mantenimento (in caso di mancanza di alimentazione)	20 ms a 120 V AC 80 ms a 240 V AC	10 ms a 24 V DC	
Fusibile interno, non sostituibile dall'utente	3 A, 250 V, lento		
Alimentazione dei sensori			
Campo di tensione	20,4 ... 28,8 V DC	L+ meno 4 V DC min.	
Corrente di uscita (max.)	400 mA (protetta da cortocircuito)		
Rumore di ondulazione max. (<10 MHz)	< 1 V da picco a picco	Come la linea di ingresso	
Isolamento (tra il circuito logico della CPU e l'alimentazione dei sensori)	Nessuno		
Ingressi digitali			
Numero di ingressi	14		
Tipo	Ad assorbimento/emissione di corrente (secondo IEC "tipo 1" se ad assorbimento di corrente)		
Tensione nominale	24 V DC a 4 mA, nominale		
Tensione continua ammessa	30 V DC max.		
Sovratensione transitoria	35 V DC per 0,5 sec.		
Segnale logico 1 (min.)	15 V DC a 2,5 mA		
Segnale logico 0 (max.)	5 V DC a 1 mA		
Isolamento (tra il campo e i circuiti logici)	500 V AC per 1 minuto		
Gruppi di isolamento	1		
Tempi di filtraggio	0,2; 0,4; 0,8; 1,6; 3,2; 6,4 e 12,8 ms (selezionabili in gruppi di 4)		
Frequenze di clock in ingresso agli HSC (max.) (segnale logico 1 = 15 ... 26 V DC)	A una fase: 100 KHz (Ia.0 ... Ia.5) e 30 KHz (Ia.6 ... Ib.5) In quadratura di fase: 80 KHz (Ia.0 ... Ia.5) e 20 KHz (Ia.6 ... Ib.5)		
Numero di ingressi ON contemporaneamente	14		
Lunghezza del cavo (metri)	500 schermato, 300 non schermato, 50 schermato per gli ingressi HSC		
Ingressi analogici			
Numero di ingressi	2		
Tipo	Tensione (asimmetrico)		
Campo	0 ... 10 V		
Campo di fondo scala (parola di dati)	0 ... 27648 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		
Campo di overshoot (parola di dati)	27,649 ... 32,511 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		
Overflow (parola di dati)	32,512 ... 32767 (vedere Rappresentazione degli ingressi analogici per la tensione (Pagina 324))		

Dati tecnici			
Modello	CPU 1214C AC/DC/relè	CPU 1214C DC/DC/relè	CPU 1214C DC/DC/DC
Risoluzione	10 bit		
Tensione di resistenza max.	35 V DC		
Livellamento	Nessuno, Debole, Medio o Forte (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere i tempi di risposta al gradino)		
Filtraggio del rumore	10, 50 o 60 Hz (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere le velocità di campionamento)		
Impedenza	≥100 KΩ		
Isolamento (tra il campo e i circuiti logici)	Nessuno		
Precisione (25°C / 0 ... 55°C)	3,0% / 3,5% del valore di fondo scala		
Reiezione in modo comune	40 dB, DC a 60 Hz		
Campo operativo del segnale	La tensione di segnale più quella di modo comune deve essere inferiore a +12 V e maggiore di -12 V		
Lunghezza del cavo (metri)	100 cavo doppio ritorto schermato		
Uscite digitali			
Numero di uscite	10		
Tipo	Relè, contatto Dry		MOSFET a stato solido
Campo di tensione	5 ... 30 V DC oppure 5 ... 250 V AC		20,4 ... 28,8 V DC
Segnale logico 1 a corrente max.	--		20 V DC min.
Segnale logico 0 con carico di 10 KΩ	--		0,1 V DC max.
Corrente (max.)	2.0 A		0.5 A
Carico delle lampade	30 W DC / 200 W AC		5 W
Resistenza in stato ON	0,2 Ω max. da nuova		0,6 Ω max.
Corrente di dispersione per punto	--		10 µA max.
Corrente di spunto	7 A con contatti chiusi		8 A per 100 ms max.
Protezione da sovraccarico	No		
Isolamento (tra il campo e i circuiti logici)	1500 V AC per 1 minuto (tra la bobina e il contatto) Nessuno (tra la bobina e i circuiti logici)		500 V AC per 1 minuto
Resistenza di isolamento	100 MΩ min. da nuova		--
Isolamento tra contatti aperti	750 V AC per 1 minuto		--
Gruppi di isolamento	2		1
Clamp per tensioni induttive	--		L+ meno 48 V DC, dissipazione di 1 W
Ritardo durante la commutazione (Qa.0 ... Qa.3)	10 ms max.		1,0 µs max., da off a on 3,0 µs max., da on a off
Ritardo durante la commutazione (Qa.4 ... Qb.1)	10 ms max.		50 µs max., da off a on 200 µs max., da on a off
Frequenza di uscita treni di impulsi (Qa.0 e Qa.2)	Sconsigliato		100 KHz max., 2 Hz min.
Tempo di vita in cicli meccanici (senza carico)	10,000,000 cicli di apertura/chiusura		--
Durata contatti con carico nominale	100,000 cicli di apertura/chiusura		--

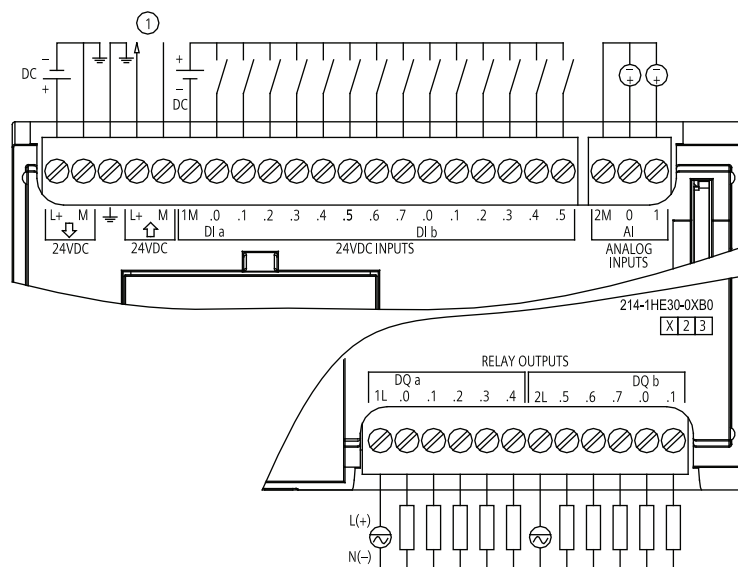
Dati tecnici			
Modello	CPU 1214C AC/DC/relè	CPU 1214C DC/DC/relè	CPU 1214C DC/DC/DC
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)		
Numero di uscite ON contemporaneamente	10		
Lunghezza del cavo (metri)	500 schermato, 150 non schermato		

Schemi elettrici



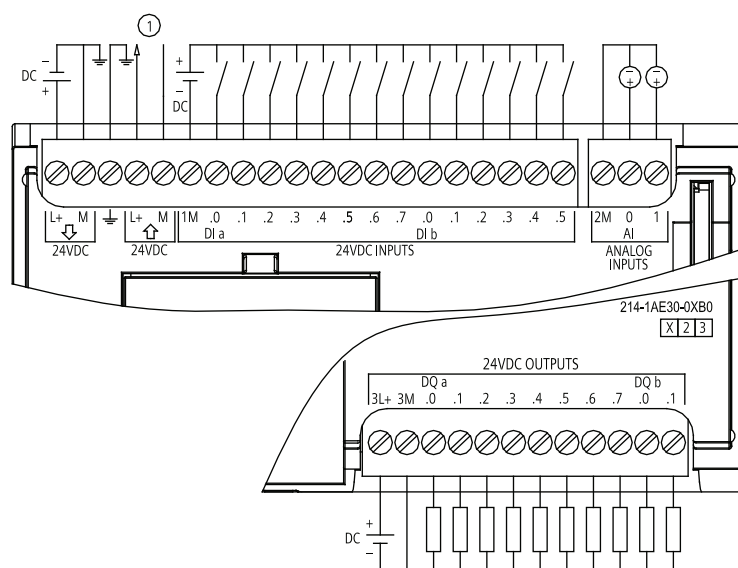
① Uscita di alimentazione per sensori a 24 V DC

Figura A-7 CPU 1214C AC/DC/relè (6ES7 214-1BE30-0XB0)



① Uscita di alimentazione per sensori a 24 V DC

Figura A-8 CPU 1214C DC/DC/relè (6ES7 214-1HE30-0XB0)



① Uscita di alimentazione per sensori a 24 V DC

Figura A-9 CPU 1214C DC/DC/DC (6ES7 214-1AE30-0XB0)

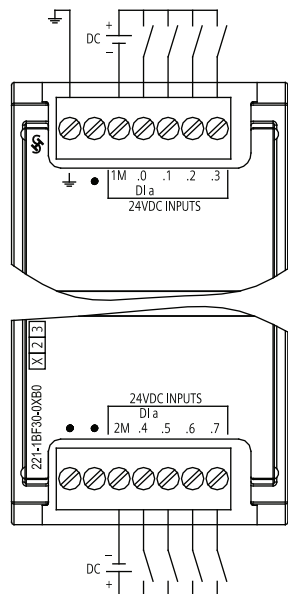
A.3 Moduli di I/O digitali (SM)

A.3.1 Dati tecnici del modulo digitale di ingresso SM 1221

Dati tecnici		
Modello	SM 1221 DI 8x24VDC	SM 1221 DI 16x24VDC
Numero di ordinazione	6ES7 221-1BF30-0XB0	6ES7 221-1BH30-0XB0
Dati generali		
Dimensioni L x A x P (mm)	45 x 100 x 75	
Peso	170 g	210 g
Dissipazione di potenza	1,5 W	2,5 W
Assorbimento di corrente (bus SM)	105 mA	130 mA
Assorbimento di corrente (24 V DC)	4 mA/ingresso utilizzato	4 mA/ingresso utilizzato
Ingressi digitali		
Numero di ingressi	8	16
Tipo	Ad assorbimento/emissione di corrente (secondo IEC "tipo 1" se ad assorbimento di corrente)	
Tensione nominale	24 V DC a 4 mA, nominale	
Tensione continua ammessa	30 V DC max.	
Sovratensione transitoria	35 V DC per 0,5 sec.	
Segnale logico 1 (min.)	15 V DC a 2,5 mA	
Segnale logico 0 (max.)	5 V DC a 1 mA	
Isolamento (tra il campo e i circuiti logici)	500 V AC per 1 minuto	
Gruppi di isolamento	2	4
Tempi di filtraggio	0,2; 0,4; 0,8; 1,6; 3,2; 6,4 e 12,8 ms (selezionabili in gruppi di 4)	
Numero di ingressi ON contemporaneamente	8	16
Lunghezza del cavo (metri)	500 schermato, 300 non schermato	

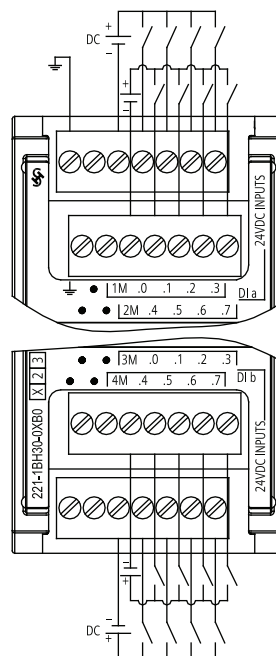
Schemi elettrici

SM 1221 DI 8 x 24 VDC



6ES7 221-1BF30-0XB0

SM 1221 DI 16 x 24 VDC



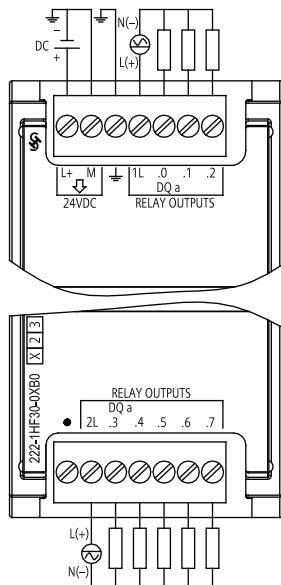
6ES7 221-1BH30-0XB0

A.3.2 Dati tecnici del modulo digitale di uscita SM 1222

Dati tecnici				
Modello	SM 1222 DQ 8xrelè	SM1222 DQ 16xrelè	SM1222 DQ 8x24VDC	SM1222 DQ 16x24VDC
Numero di ordinazione	6ES7 222-1HF30-0XB0	6ES7 222-1HH30-0XB0	6ES7 222-1BF30-0XB0	6ES7 222-1BH30-0XB0
Dati generali				
Dimensioni L x A x P (mm)	45 x 100 x 75			
Peso	190 g	260 g	180 g	220 g
Dissipazione di potenza	4,5 W	8,5 W	1,5 W	2,5 W
Assorbimento di corrente (bus SM)	120 mA	135 mA	120 mA	140 mA
Assorbimento di corrente (24 V DC)	11 mA / con bobina relè		--	
Uscite digitali				
Numero di uscite	8	16	8	16
Tipo	Relè, contatto Dry		MOSFET a stato solido	
Campo di tensione	5 ... 30 V DC oppure 5 ... 250 V AC		20,4 ... 28,8 V DC	
Segnale logico 1 a corrente max.	--		20 V DC min.	
Segnale logico 0 con carico di 10 KΩ	--		0,1 V DC max.	
Corrente (max.)	2,0 A		0,5 A	
Carico delle lampade	30 W DC / 200 W AC		5 W	
Resistenza contatto in stato ON	0,2 Ω max. da nuova		0,6 Ω max.	
Corrente di dispersione per punto	--		10 µA max.	
Corrente di spunto	7 A con contatti chiusi		8 A per 100 ms max.	
Protezione da sovraccarico	No			
Isolamento (tra il campo e i circuiti logici)	1500 V AC per 1 minuto (tra la bobina e il contatto) Nessuno (tra la bobina e i circuiti logici)		500 V AC per 1 minuto	
Resistenza di isolamento	100 MΩ min. da nuova		--	
Isolamento tra contatti aperti	750 V AC per 1 minuto		--	
Gruppi di isolamento	2	4	1	1
Corrente per comune (max.)	10 A		4 A	8 A
Clamp per tensioni induttive	--		L+ meno 48 V, dissipazione di 1 W	
Ritardo durante la commutazione	10 ms max.		50 µs max. da off a on 200 µs max. da on a off	
Tempo di vita in cicli meccanici (senza carico)	10,000,000 cicli di apertura/chiusura		--	
Durata contatti con carico nominale	100,000 cicli di apertura/chiusura		--	
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)			
Numero di uscite ON contemporaneamente	8	16	8	16
Lunghezza del cavo (metri)	500 schermato, 150 non schermato			

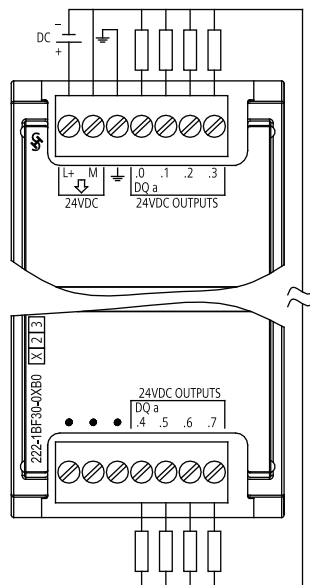
Schemi elettrici

SM 1222 DQ 8 x relè



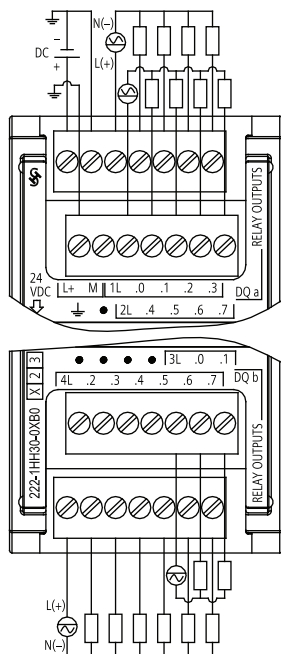
6ES7 222-1HF30-0XB0

SM 1222 DQ 8 x 24 VDC



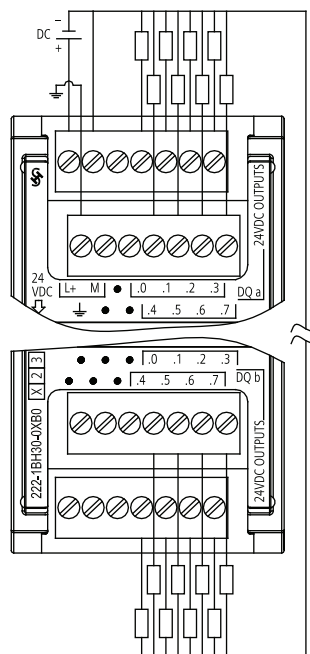
6ES7 222-1BF30-0XB0

SM 1222 DQ 16 x relè



6ES7 222-1HH30-0XB0

SM 1222 DQ 16 x 24 VDC



6ES7 222-1BH30-0XB0

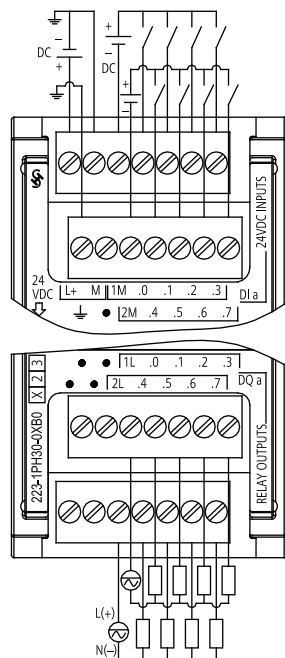
A.3.3 Dati tecnici del modulo digitale di ingresso/uscita SM 1223

Dati tecnici				
Modello	SM 1223 DI 8x24 VDC, DQ 8xrelè	SM 1223 DI 16x24 VDC, DQ 16xrelè	SM 1223 DI 8x24 VDC, DQ 8x24 VDC	SM 1223 DI 16x24 VDC, DQ16x24 VDC
Numero di ordinazione	6ES7 223-1PH30-0XB0	6ES7 223-1PL30-0XB0	6ES7 223-1BH30-0XB0	6ES7 223-1BL30-0XB0
Dimensioni L x A x P (mm)	45 x 100 x 75	70 x 100 x 75	45 x 100 x 75	70 x 100 x 75
Peso	230 g	350 g	210 g	310 g
Dissipazione di potenza	5,5 W	10 W	2,5 W	4,5 W
Assorbimento di corrente (bus SM)	145 mA	180 mA	145 mA	185 mA
Assorbimento di corrente (24 V DC)	4 mA/ingresso utilizzato 11 mA / con bobina relè		4 mA/ingresso utilizzato	
Ingressi digitali				
Numero di ingressi	8	16	8	16
Tipo	Ad assorbimento/emissione di corrente (secondo IEC "tipo 1" se ad assorbimento di corrente)			
Tensione nominale	24 V DC a 4 mA, nominale			
Tensione continua ammessa	30 V DC max.			
Sovratensione transitoria	35 V DC per 0,5 sec.			
Segnale logico 1 (min.)	15 V DC a 2,5 mA			
Segnale logico 0 (max.)	5 V DC a 1 mA			
Isolamento (tra il campo e i circuiti logici)	500 V AC per 1 minuto			
Gruppi di isolamento	2	2	2	2
Tempi di filtraggio	0,2; 0,4; 0,8; 1,6; 3,2; 6,4 e 12,8 ms, selezionabili in gruppi di 4			
Numero di ingressi ON contemporaneamente	8	16	8	16
Lunghezza del cavo (metri)	500 schermato, 300 non schermato			
Uscite digitali				
Numero di uscite	8	16	8	16
Tipo	Relè, contatto Dry		MOSFET a stato solido	
Campo di tensione	5 ... 30 V DC oppure 5 ... 250 V AC		20,4 ... 28,8 V DC	
Segnale logico 1 a corrente max.	--		20 V DC min.	
Segnale logico 0 con carico di 10 KΩ	--		0,1 V DC max.	
Corrente (max.)	2,0 A		0,5 A	
Carico delle lampade	30 W DC / 200 W AC		5 W	
Resistenza contatto in stato ON	0,2 Ω max. da nuova		0,6 Ω max.	
Corrente di dispersione per punto	--		10 µA max.	
Corrente di spunto	7 A con contatti chiusi		8 A per 100 ms max.	
Protezione da sovraccarico	No			

Dati tecnici				
Modello	SM 1223 DI 8x24 VDC, DQ 8xrelè	SM 1223 DI 16x24 VDC, DQ 16xrelè	SM 1223 DI 8x24 VDC, DQ 8x24 VDC	SM 1223 DI 16x24 VDC, DQ16x24 VDC
Isolamento (tra il campo e i circuiti logici)	1500 V AC per 1 minuto (tra la bobina e il contatto) Nessuno (tra la bobina e i circuiti logici)		500 V AC per 1 minuto	
Resistenza di isolamento	100 MΩ min. da nuova		--	
Isolamento tra contatti aperti	750 V AC per 1 minuto		--	
Gruppi di isolamento	2	4	1	1
Corrente per comune	10A	8 A	4 A	8 A
Clamp per tensioni induttive	--		L+ meno 48 V, dissipazione di 1 W	
Ritardo durante la commutazione	10 ms max.		50 μs max., da off a on 200 μs max., da on a off	
Tempo di vita in cicli meccanici (senza carico)	10,000,000 cicli di apertura/chiusura		--	
Durata contatti con carico nominale	100,000 cicli di apertura/chiusura		--	
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)			
Numero di uscite ON contemporaneamente	8	16	8	16
Lunghezza del cavo (metri)	500 schermato, 150 non schermato			

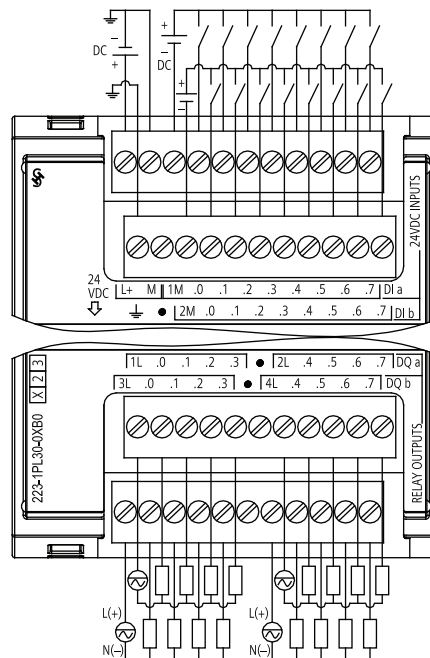
Schemi elettrici

SM 1223 DI 8 x 24 VDC, DQ 8 x relè



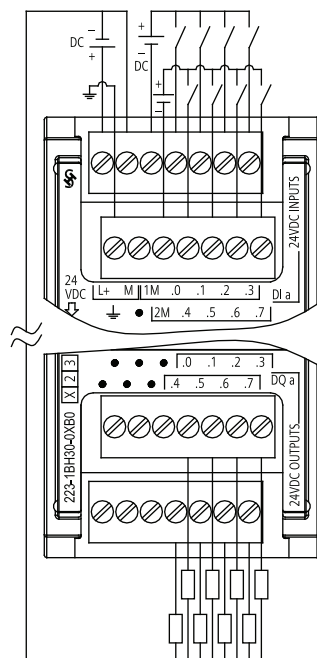
6ES7 223-1PH30-0XB0

SM1223 DI 16 x 24 VDC, DQ 16 x relè



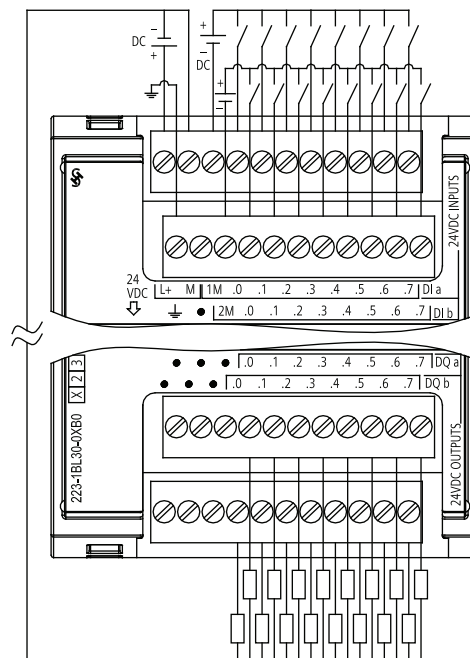
6ES7 223-1PL30-0XB0

SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC



6ES7 223-1BH30-0XB0

SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC



6ES7 223-1BL30-0XB0

A.4 Moduli di I/O analogici (SM)

A.4.1 Dati tecnici dei moduli analogici SM 1231, SM 1232, SM 1234

Dati tecnici			
Modello	SM 1231 AI 4x13bit	SM 1231 AI 8x13bit	SM 1234 AI 4x13bit AQ 2x14bit
Numero di ordinazione	6ES7 231-4HD30-0XB0	6ES7 231-4HF30-0XB0	6ES7 234-4HE30-0XB0
Dati generali			
Dimensioni L x A x P (mm)	45 x 100 x 75	45 x 100 x 75	45 x 100 x 75
Peso	180 g	180 g	220 g
Dissipazione di potenza	1.5 W	1,5 W	2.0 W
Assorbimento di corrente (bus SM)	80 mA	90 mA	80 mA
Assorbimento di corrente (24 V DC)	45 mA	45 mA	60 mA (senza carico)
Ingressi analogici			
Numero di ingressi	4	8	4
Tipo	Tensione o corrente (differenziale): Selezionabili in gruppi di 2		
Campo	± 10 V, ± 5 V, $\pm 2,5$ V o 0 ... 20 mA		
Campo di fondo scala (parola di dati)	-27,648 ... 27,648		
Campo di overshoot/undershoot (parola di dati)	Tensione: 32,511 ... 27,649 / -27,649 ... -32,512 Corrente: 32,511 ... 27,649 / 0 ... -4864 (vedere Rappresentazione degli ingressi analogici per la tensione e Rappresentazione degli ingressi analogici per la corrente (Pagina 324))		
Overflow/underflow (parola di dati)	Tensione: 32,767 ... 32,512 / -32,513 ... -32,768 Corrente: 32,767 ... 32,512 / -4865 ... -32,768 (vedere Rappresentazione degli ingressi analogici per la tensione e Rappresentazione degli ingressi analogici per la corrente (Pagina 324))		
Risoluzione	12 bit + bit di segno		
Tensione/corrente di resistenza max.	± 35 V / ± 40 mA		
Livellamento	Nessuno, Debole, Medio o Forte (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere i tempi di risposta al gradino)		
Filtraggio del rumore	400, 60, 50 o 10 Hz (vedere Tempi di risposta degli ingressi analogici (Pagina 324) per conoscere le velocità di campionamento)		
Impedenza	≥ 9 M Ω (tensione) / 250 Ω (corrente)		
Isolamento (tra il campo e i circuiti logici)	Nessuno		
Precisione (25°C / 0 ... 55°C)	$\pm 0,1\%$ / $\pm 0,2\%$ del valore di fondo scala		
Tempo di conversione da analogico a digitale	625 μ s (reiezione di 400 Hz)		
Reiezione in modo comune	40 dB, DC a 60 Hz		
Campo operativo del segnale	La tensione di segnale più quella di modo comune deve essere inferiore a +12 V e maggiore di -12 V		
Lunghezza del cavo (metri)	100 metri, cavo doppio ritorto schermato		

Dati tecnici			
Modello	SM 1231 AI 4x13bit	SM 1231 AI 8x13bit	SM 1234 AI 4x13bit AQ 2x14bit
Diagnostica			
Overflow/underflow	Sì ¹	Sì ¹	Sì ¹
Cortocircuito verso terra (solo nel modo in tensione)	Non applicabile	Non applicabile	Sì nelle uscite
Rottura conduttore (solo nel modo in corrente)	Non applicabile	Non applicabile	Sì nelle uscite
Tensione bassa 24 V DC	Sì	Sì	Sì

¹ Se si applica all'ingresso una tensione superiore a +30 V DC o inferiore a -15 V DC il valore risultante sarà sconosciuto e l'overflow o l'underflow corrispondenti potrebbero non essere attivi.

Dati tecnici			
Modello	SM 1232 AQ 2x14bit	SM 1232 AQ 4x14bit	SM 1234 AI 4x13bit AQ 2x14bit
Numero di ordinazione	6ES7 232-4HB30-0XB0	6ES7 232-4HD30-0XB0	6ES7 234-4HE30-0XB0
Dati generali			
Dimensioni L x A x P (mm)	45 x 100 x 75	45 x 100 x 75	45 x 100 x 75
Peso	180 g	180 g	220 g
Dissipazione di potenza	1.5 W	1.5 W	2.0 W
Assorbimento di corrente (bus SM)	80 mA	80 mA	80 mA
Assorbimento di corrente (24 V DC)	45 mA (senza carico)	45 mA (senza carico)	60 mA (senza carico)
Uscite analogiche			
Numero di uscite	2	4	2
Tipo	Tensione o corrente		
Campo	±10 V o 0 ... 20 mA		
Risoluzione	Tensione: 14 bit; Corrente: 13 bit		
Campo di fondo scala (parola di dati)	Tensione: -27,648 ... 27,648 ; Corrente: 0 ... 27,648 (vedere Rappresentazione delle uscite analogiche per la tensione e Rappresentazione delle uscite analogiche per la corrente) (Pagina 324)		
Precisione (25°C / 0 ... 55°C)	±0,3% / ±0,6% del valore di fondo scala		
Tempo di assestamento (95% del nuovo valore)	Tensione: 300 µS (R), 750 µS (1 uF); Corrente: 600 µS (1 mH), 2 ms (10 mH)		
Impedenza di carico	Tensione: ≥ 1000 Ω; Corrente: ≤ 600 Ω		
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)		
Isolamento (tra il campo e i circuiti logici)	Nessuno		
Lunghezza del cavo (metri)	100 metri, cavo doppio ritorto schermato		
Diagnostica			
Overflow/underflow	Sì	Sì	Sì ¹
Cortocircuito verso terra (solo nel modo in tensione)	Sì	Sì	Sì nelle uscite

Dati tecnici			
Modello	SM 1232 AQ 2x14bit	SM 1232 AQ 4x14bit	SM 1234 AI 4x13bit AQ 2x14bit
Rottura conduttore (solo nel modo in corrente)	Sì	Sì	Sì nelle uscite
Tensione bassa 24 V DC	Sì	Sì	Sì

- ¹ Se si applica all'ingresso una tensione superiore a +30 V DC o inferiore a -15 V DC il valore risultante sarà sconosciuto e l'overflow o l'underflow corrispondenti potrebbero non essere attivi.

Tempi di risposta degli ingressi analogici

Risposta al gradino dei moduli analogici SM (ms)				
0 V ... 10 V misurata al 95%				
Livellamento	Frequenza reiezione:			
	400 Hz	60 Hz	50 Hz	10 Hz
Nessuno	4	18	22	100
Debole	9	52	63	320
Medio	32	203	241	1200
Forte	61	400	483	2410
Frequenza di campionamento	• 0.625	• 4.17	• 5	• 25
• 4 canali	• 1.25	• 4.17	• 5	• 25
• 8 canali				

Risposta al gradino degli ingressi analogici della CPU (ms)			
0 V ... 10 V misurata al 95%			
Livellamento	Frequenza reiezione:		
	60 Hz	50 Hz	10 Hz
Nessuno	63	65	130
Debole	84	93	340
Medio	221	258	1210
Forte	424	499	2410
Frequenza di campionamento	4.17	5	25

Rappresentazione degli ingressi analogici per la tensione

Sistema	Campo di misura della tensione						
Decimale	Esadecimale	±10 V	±5 V	±2,5 V		0 ... 10 V	
32767	7FFF	11,851 V	5,926 V	2,963 V	Overflow	11,851V	Overflow
32512	7F00						
32511	7EFF	11,759 V	5,879 V	2,940 V	Campo di overshoot	11,759 V	Campo di overshoot
27649	6C01						
27648	6C00	10 V	5 V	2,5 V	Campo nominale	10 V	Campo nominale
20736	5100	7,5 V	3,75 V	1,875 V		7,5 V	
1	1	361,7 μV	180,8 μV	90,4 μV		361,7 μV	
0	0	0 V	0 V	0 V		0 V	
-1	FFFF					I valori negativi non sono ammessi	
-20736	AF00	-7,5 V	-3,75 V	-1,875 V			
-27648	9400	-10 V	-5 V	-2,5 V			
-27649	93FF				Campo di undershoot		
-32512	8100	-11,759 V	-5,879 V	-2,940 V			
-32513	80FF				Underflow		
-32768	8000	-11.851 V	-5.926 V	-2.963 V			

Rappresentazione degli ingressi analogici per la corrente

Sistema	Campo di misura della corrente		
Decimale	Esadecimale	da 0 mA a 20 mA	
32767	7FFF	23,70 mA	Overflow
32512	7F00		
32511	7EFF	23,52 mA	Campo di overshoot
27649	6C01		
27648	6C00	20 mA	Campo nominale
20736	5100	15 mA	
1	1	723,4 nA	
0	0	0 mA	
-1	FFFF		Campo di undershoot
-4864	ED00	-3,52 mA	
-4865	ECFF		Underflow
-32768	8000		

Rappresentazione delle uscite analogiche per la tensione

Sistema		Campo della tensione in uscita	
Decimale	Esadecimale	± 10 V	
32767	7FFF	Vedere la nota 1	Overflow
32512	7F00	Vedere la nota 1	
32511	7EFF	11,76 V	Campo di overshoot
27649	6C01		
27648	6C00	10 V	Campo nominale
20736	5100	7,5 V	
1	1	361,7 μ V	
0	0	0 V	
-1	FFFF	-361,7 μ V	
-20736	AF00	-7,5 V	
-27648	9400	-10 V	
-27649	93FF		Campo di undershoot
-32512	8100	-11,76 V	
-32513	80FF	Vedere la nota 1	Underflow
-32768	8000	Vedere la nota 1	

- ¹ In una condizione di overflow o underflow, le uscite analogiche si comportano in base alle proprietà di configurazione impostate per il modulo di I/O analogici. Nel parametro "Comportamento in caso di STOP della CPU" selezionare: Imposta valore sostitutivo o Mantieni ultimo valore.

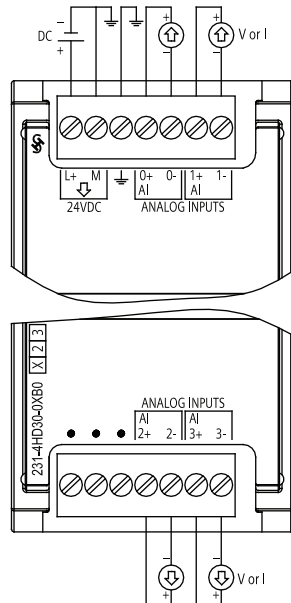
Rappresentazione delle uscite analogiche per la corrente

Sistema		Campo della corrente in uscita	
Decimale	Esadecimale	± 20 mA	
32767	7FFF	Vedere la nota 1	Overflow
32512	7F00	Vedere la nota 1	
32511	7EFF	23.52 mA	Campo di overshoot
27649	6C01		
27648	6C00	20 mA	Campo nominale
20736	5100	15 mA	
1	1	723,4 nA	
0	0	0 mA	
-1	FFFF		
-32512	8100		Campo di undershoot
-32513	80FF	Vedere la nota 1	
-32768	8000	Vedere la nota 1	Underflow

1. In una condizione di overflow o underflow, le uscite analogiche si comportano in base alle proprietà di configurazione impostate per il modulo di I/O analogici. Nel parametro "Comportamento in caso di STOP della CPU" selezionare: Imposta valore sostitutivo o Mantieni ultimo valore.

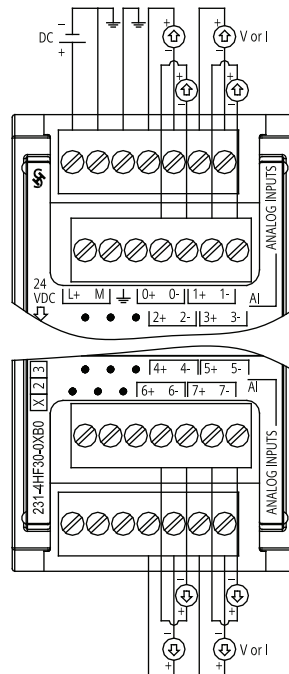
Schemi elettrici

SM 1231 AI 4 x 13 bit



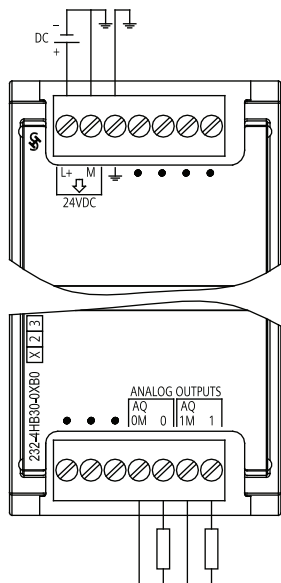
6ES7 231-4HD30-0XB0

SM 1231 AI 8 x 13 bit



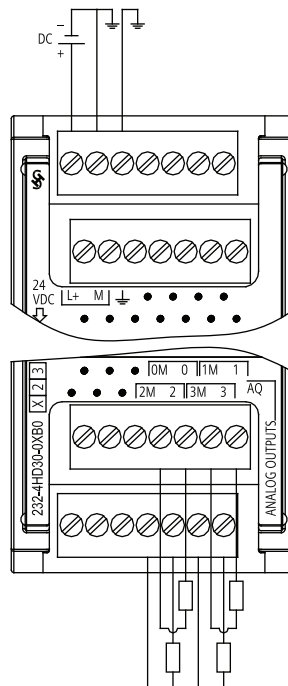
6ES7 231-4HF30-0XB0

SM 1232 AQ 2 x 14 bit



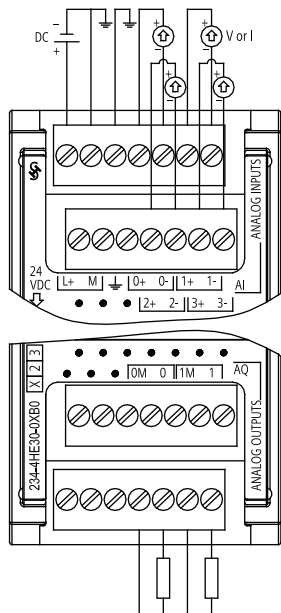
6ES7 232-4HB30-0XB0

SM 1232 AQ 4 x 14 bit



6ES7 232-4HD30-0XB0

SM 1234 AI 4 x 13 Bit / AQ 2 x 14 bit



6ES7 234-4HE30-0XB0

A.5 Signal board (SB)

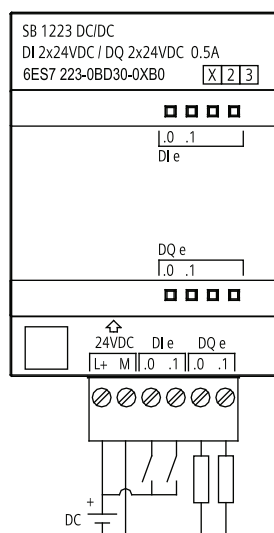
A.5.1 Dati tecnici dell'SB 1223 2 ingressi 24 V DC / 2 uscite 24 V DC

Dati tecnici delle signal board digitali

Dati tecnici	
Modello	SB 1223 DI 2x24VDC, DQ 2x24VDC
Numero di ordinazione	6ES7 223-0BD30-0XB0
Dati generali	
Dimensioni L x A x P (mm)	38 x 62 x 21
Peso	40 g
Dissipazione di potenza	1,0 W
Assorbimento di corrente (bus SM)	50 mA
Assorbimento di corrente (24 V DC)	4 mA/ingresso utilizzato
Ingressi digitali	
Numero di ingressi	2
Tipo	IEC "tipo 1" se ad assorbimento di corrente
Tensione nominale	24 V DC a 4 mA, nominale
Tensione continua ammessa	30 V DC max.
Sovratensione transitoria	35 V DC per 0,5 sec.
Segnale logico 1 (min.)	15 V DC a 2,5 mA
Segnale logico 0 (max.)	5 V DC a 1 mA
Frequenze di clock in ingresso agli HSC (max.)	20 kHz (15 ... 30 V DC) 30 kHz (15 ... 26 V DC)
Isolamento (tra il campo e i circuiti logici)	500 V AC per 1 minuto
Gruppi di isolamento	1
Tempi di filtraggio	0,2; 0,4; 0,8; 1,6; 3,2; 6,4 e 12,8 ms selezionabili in gruppi di 2
Numero di ingressi ON contemporaneamente	2
Lunghezza del cavo (metri)	500 schermato, 300 non schermato
Uscite digitali	
Numero di uscite	2
Tipo di uscite	MOSFET a stato solido
Campo di tensione	20,4 ... 28,8 V DC
Segnale logico 1 a corrente max.	20 V DC min.
Segnale logico 0 con carico di 10 K Ω	0,1 V DC max.
Corrente (max.)	0,5 A
Carico delle lampade	5 W
Resistenza contatto in stato ON	0,6 Ω max.
Corrente di dispersione per punto	10 μ A max.
Frequenza di uscita treni di impulsi	20 KHz max., 2 Hz min.

Dati tecnici	
Modello	SB 1223 DI 2x24VDC, DQ 2x24VDC
Corrente di spunto	5 A per 100 ms max.
Protezione da sovraccarico	No
Isolamento (tra il campo e i circuiti logici)	500 V AC per 1 minuto
Gruppi di isolamento	1
Correnti per comune	1 A
Clamp per tensioni induttive	L+ meno 48 V, dissipazione di 1 W
Ritardo durante la commutazione	2 μ s max., da off a on 10 μ s max., da on a off
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)
Numero di uscite ON contemporaneamente	2
Lunghezza del cavo (metri)	500 schermato, 150 non schermato

Schema elettrico dell'SB 1223 2 ingressi 24 V DC / 2 uscite 24 V DC

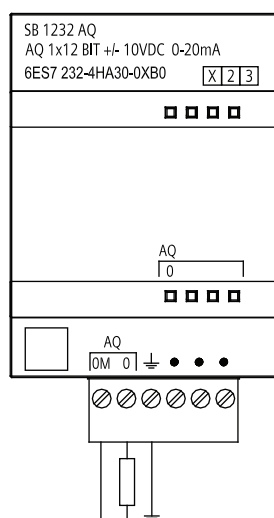


A.5.2 Dati tecnici dell'SB 1232 1 uscita analogica

Dati tecnici delle signal board analogiche

Dati tecnici	
Modello	SB 1223 AQ 1x12bit
Numero di ordinazione	6ES7 232-4HA30-0XB0
Dati generali	
Dimensioni L x A x P (mm)	38 x 62 x 21 mm
Peso	40 g
Dissipazione di potenza	1,5 W
Assorbimento di corrente (bus SM)	15 mA
Assorbimento di corrente (24 V DC)	40 mA (senza carico)
Uscite analogiche	
Numero di uscite	1
Tipo	Tensione o corrente
Campo	± 10 V o 0 ... 20 mA
Risoluzione	Tensione: 12 bit Corrente: 11 bit
Campo di fondo scala (parola di dati)	Tensione: -27.648 ... 27.648 Corrente: Da 0 a 27,648
Precisione (25°C / 0 ... 55°C)	$\pm 0.5\%$ / $\pm 1\%$ del valore di fondo scala
Tempo di assestamento (95% del nuovo valore)	Tensione: 300 μ S (R), 750 μ S (1 uF) Corrente: 600 μ S (1 mH), 2 ms (10 mH)
Impedenza di carico	Tensione: $\geq 1000 \Omega$ Corrente: $\leq 600 \Omega$
Comportamento in caso di commutazione RUN - STOP	Ultimo valore o valore sostitutivo (valore di default 0)
Isolamento (tra il campo e i circuiti logici)	Nessuno
Lunghezza del cavo (metri)	100 metri, cavo doppio ritorto schermato
Diagnostica	
Overflow/underflow	Sì
Cortocircuito verso terra (solo nel modo in tensione)	Sì
Rottura conduttore (solo nel modo in corrente)	Sì

Schema elettrico dell'SB 1232 1 uscita analogica

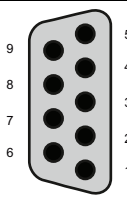


A.6 Moduli di comunicazione (CM)

A.6.1 CM 1241 RS485, dati tecnici

Tabella A- 1 Modulo di comunicazione CM 1241 RS485

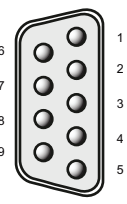
Dati tecnici	
Numero di ordinazione	6ES7 241-1CH30-0XB0
Dimensioni e peso	
Dimensioni	30 x 100 x 75 mm
Peso	150 g
Trasmettitore e ricevitore	
Campo della tensione in modo comune	-7 V ... +12 V, 1 secondo, 3 V RMS continuo
Tensione di uscita differenziale trasmettitore	2 V min. a $R_L=100\ \Omega$ 1,5 V min. a $R_L=54\ \Omega$
Resistenze di terminazione e polarizzazione	10 K Ω ... +5 V su B, PROFIBUS piedino 3 10 K Ω ... GND su A, PROFIBUS piedino 8
Impedenza di ingresso ricevitore	5,4 K Ω min. compresa la terminazione
Soglia/sensibilità ricevitore	+/- 0,2 V min., 60 mV isteresi tipica
Isolamento Tra il segnale RS485 e la massa del telaio Tra il segnale RS485 e il comune dei circuiti logici della CPU	500 V AC per 1 minuto
Lunghezza conduttore schermato	1000 m max.
Alimentazione	
Dissipazione di potenza	1,1 W
Da +5 V DC	220 mA

Piedino	Descrizione	Connettore (femmina)	Piedino	Descrizione
1 GND	Massa logica o di comunicazione		6 PWR	+5 V con resistenza in serie di 100 ohm: Uscita
2	Non collegata		7	Non collegata
3 TxD+	Segnale B (RxD/TxD+): Ingresso/uscita		8 TXD-	Segnale A (RxD/TxD-): Ingresso/uscita
4 RTS	Request to send (richiesta di invio) (livello TTL): Uscita		9	Non collegata
5 GND	Massa logica o di comunicazione		SHELL	Massa del telaio

A.6.2 CM 1241 RS232, dati tecnici

Modulo di comunicazione CM 1241 RS232

Dati tecnici	
Numero di ordinazione	6ES7 241-1AH30-0XB0
Dimensioni e peso	
Dimensioni	30 x 100 x 75 mm
Peso	150 g
Trasmettitore e ricevitore	
Tensione di uscita trasmettitore	+/- 5 V min. a $R_L=3\text{ K } \Omega$
Tensione di uscita trasmettitore	+/- 15 V DC max.
Impedenza di ingresso ricevitore	3 K Ω min.
Soglia/sensibilità ricevitore	0,8 V min. bassa, 2,4 V max. alta, 0,5 V isteresi tipica
Tensione di ingresso ricevitore	+/- 30 V DC max.
Isolamento Tra il segnale RS 232 e la massa del telaio Tra il segnale RS 232 e il comune dei circuiti logici della CPU	500 V AC per 1 minuto
Lunghezza conduttore schermato	10 m max.
Alimentazione	
Dissipazione di potenza	1,1 W
Da +5 V DC	220 mA

Piedino	Descrizione	Connettore (maschio)	Piedino	Descrizione
1 DCD	Data carrier detect (rileva portante): Ingresso		6 DSR	Data set ready (set di dati pronto): Ingresso
2 RxD	Dati ricevuti da DCE: Ingresso		7 RTS	Request to Send (richiesta di invio): Uscita
3 TxD	Dati trasmessi da DCE: Uscita		8 CTS	Clear to send (pronto a trasmettere): Ingresso
4 DTR	Data terminal ready (terminale dati pronto): Uscita		9 RI	Ring indicator (indicatore di squillo) (non utilizzato)
5 GND	Massa logica		SHELL	Massa del telaio

A.7 Memory card SIMATIC

Dati tecnici delle memory card

Numero di ordinazione	Capacità
6ES7 954-8LF00-0AA0	24 MB
6ES7 954-8LB00-0AA0	2 MB

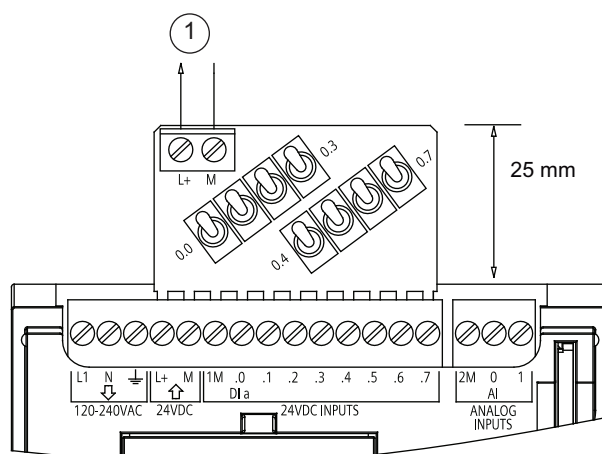
A.8 Simulatori di ingressi

Modello	Simulatore a 8 posizioni	Simulatore a 14 posizioni
Numero di ordinazione	6ES7 274-1XF30-0XA0	6ES7 274-1XH30-0XA0
Dimensioni L x A x P (mm)	43 x 35 x 23	67 x 35 x 23
Peso	20 g	30 g
I/O	8	14
Utilizzato con la CPU	CPU 1211C, CPU 1212C	CPU 1214C

AVVERTENZA

Questi simulatori di ingressi non sono stati certificati per l'utilizzo in luoghi pericolosi della Classe I DIV 2 e della Classe I Zona 2, perché gli switch possono provocare scintille ed esplosioni se utilizzati negli ambienti di questo tipo.

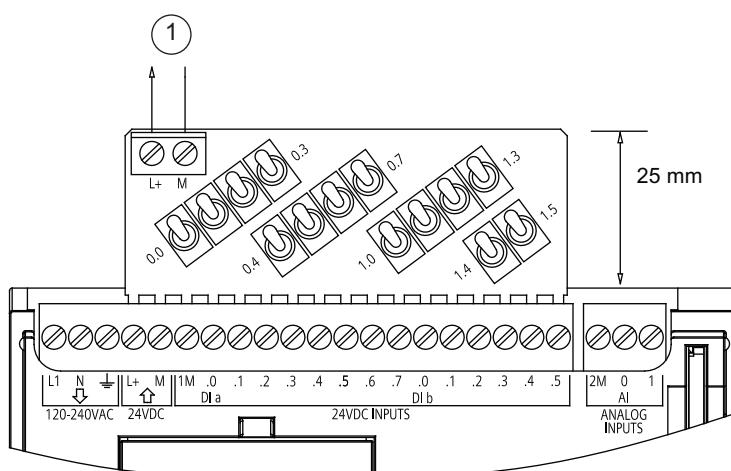
Simulatore a 8 posizioni



① Uscita di alimentazione sensori a 24 V DC

6ES7 274-1XF30-0XA0

Simulatore a 14 posizioni



① Uscita di alimentazione sensori a 24 V DC

6ES7 274-1XH30-0XA0

A.9 Prolunga per I/O

Dati tecnici	
Numero di ordinazione	6ES7 290-6AA30-0XA0
Lunghezza del cavo	2 m
Peso	200 g

La prolunga per gli I/O dispone di connettore maschio e femmina.

1. Collegare il connettore maschio al connettore di bus sul lato destro del modulo di I/O
2. e il connettore femmina al connettore di bus sul lato sinistro.
 - Inserire l'occhiello del connettore femmina nel connettore di bus del case.
 - Inserire il connettore femmina nel connettore di bus.

Calcolo del budget di potenza

La CPU dispone di un alimentatore integrato che provvede all'alimentazione della CPU stessa, dei moduli di ampliamento e delle altre utenze che richiedono un'alimentazione a 24 V DC.

Sono disponibili tre tipi di moduli di ampliamento:

- I moduli di I/O (SM) vanno montati a destra della CPU. Ogni CPU consente di collegare un numero particolare di moduli indipendentemente dal budget di potenza.
 - La CPU 1214 consente di collegare 8 moduli di I/O
 - La CPU 1212 consente di collegare 2 moduli di I/O
 - La CPU 1211 non consente di collegare alcun modulo
- I moduli di comunicazione (CM) vanno montati a sinistra della CPU. È consentito collegare al massimo 3 moduli per qualsiasi CPU indipendentemente dal budget di potenza.
- Le signal board (SB) vanno montate nel lato superiore della CPU. Ogni CPU può contenerne al massimo una.

Le informazioni fornite di seguito possono essere utilizzate come guida per determinare quanta potenza (o corrente) la CPU è in grado di erogare alla propria configurazione.

Ogni CPU eroga una corrente continua a 5 V DC e 24 V DC.

- La CPU fornisce inoltre una tensione a 5V DC ai moduli di ampliamento (se presenti). Se la tensione a 5 V DC assorbita dai moduli di ampliamento supera il budget di potenza della CPU si devono rimuovere i moduli di ampliamento finché la corrente assorbita non rientra nel budget disponibile.
- Ogni CPU dispone di un'alimentazione per sensori che fornisce una tensione a 24 V DC agli ingressi locali o alle bobine dei relè dei moduli di ampliamento. Se la potenza richiesta a 24 V DC è superiore al budget di potenza della CPU, è necessario aggiungere un alimentatore esterno a 24 V DC per alimentare i moduli di ampliamento. L'alimentatore a 24 V DC deve essere collegato manualmente agli ingressi e alle bobine del relè.

AVVERTENZA

Se si collega un alimentatore esterno a 24 V DC in parallelo all'alimentazione per sensori DC può verificarsi un conflitto tra le due alimentazioni che cercheranno di imporre il proprio livello di tensione di uscita preferenziale.

Ne potrebbero derivare una riduzione della durata o il guasto immediato di uno o entrambi gli alimentatori, con conseguente funzionamento imprevedibile del sistema. Ciò può causare la morte o gravi lesioni alle persone e/o danni alle cose.

L'alimentazione per sensori DC nella CPU e le sorgenti di alimentazione esterna devono alimentare I/O diversi. È ammesso un unico collegamento dei poli comuni.

Alcune porte di ingresso dell'alimentazione a 24 V del sistema PLC sono interconnesse, ovvero un circuito logico comune collega tra loro più morsetti M. Sono esempi di circuiti interconnessi, se contrassegnati come "non isolati" nelle schede tecniche, l'ingresso di

alimentazione a 24 V DC della CPU, l'ingresso di alimentazione della bobina del relè di un SM e un ingresso di alimentazione analogico non isolato. Tutti i morsetti M non isolati devono essere collegati allo stesso potenziale di riferimento esterno.



AVVERTENZA

Se si collegano i morsetti M non isolati a potenziali di riferimento diversi si formano flussi di corrente indesiderati che possono danneggiare il PLC e le apparecchiature a cui è collegato o farli funzionare in modo imprevedibile.

Ciò può causare la morte o gravi lesioni alle persone e/o danni alle cose.

È quindi importante accertarsi che i morsetti M non isolati del sistema PLC siano collegati allo stesso potenziale di riferimento.

Per informazioni sul budget di potenza delle CPU e sulla potenza richiesta dai moduli di ampliamento consultare i dati tecnici (Pagina 295).

Nota

Se si supera il budget di potenza della CPU potrebbe non essere possibile collegare il numero massimo di moduli.

B.1 Esempio di calcolo della potenza richiesta

La tabella sotto riportata illustra un esempio di calcolo della potenza richiesta da un PLC costituito da una CPU 1214C AC/DC/relè, tre SM 1223 8 ingressi DC/8 uscite relè e un SM 1221 8 ingressi DC. Questo esempio dispone complessivamente di 46 ingressi e 34 uscite.

Nota

La CPU ha già assegnato la potenza necessaria per le bobine relè interne per cui non è necessario tenerne conto nel calcolo del budget di potenza.

In questo esempio la tensione a 5V DC fornita dalla CPU è sufficiente ad alimentare gli SM, mentre l'alimentazione per sensori a 24 V DC risulta insufficiente ad alimentare gli ingressi e le bobine dei relè. Gli I/O richiedono 448 mA ma la CPU fornisce solo 400 mA. Per alimentare tutti gli ingressi e le uscite a 24 V DC dell'installazione è quindi necessario un ulteriore alimentatore da min. 48 mA.

Budget di potenza della CPU	5 V DC	24 V DC
CPU 1214C AC/DC/relè	1600 mA	400 mA
<i>Meno</i>		
Requisiti del sistema	5 V DC	24 V DC
CPU 1214C, 14 ingressi	-	14 * 4 mA = 56 mA
3 SM 1223, 5 V	3 * 145 mA = 435 mA	-
1 SM 1221, 5 V	1 * 105 mA = 105 mA	-
3 SM 1223, ognuno con 8 ingressi	-	3 * 8 * 4 mA = 96 mA
3 SM 1223, ognuno con 8 bobine relè	-	3 * 8 * 11 mA = 264 mA
1 SM 1221, 8 ingressi	-	8 * 4 mA = 32 mA
Fabbisogno totale	540 mA	448 mA
<i>Uguale</i>		
Bilancio di corrente	5 V DC	24 V DC
Corrente complessiva	1060 mA	(48 mA)

B.2 Calcolo della potenza richiesta

La tabella qui raffigurata può essere utilizzata per calcolare la potenza (o corrente) che la CPU S7-1200 è in grado di erogare alla propria configurazione. Per maggiori informazioni sul budget di potenza della propria CPU e sui requisiti di potenza dei moduli di I/O utilizzati, consultare i dati tecnici (Pagina 295).

Budget di potenza della CPU	5 V DC	24 V DC
<i>Meno</i>		
Requisiti del sistema	5 V DC	24 V DC
Fabbisogno totale		
<i>Uguale</i>		
Bilancio di corrente	5 V DC	24 V DC
Corrente complessiva		

Numeri di ordinazione

CPU		Numero di ordinazione
CPU 1211C	CPU 1211C DC/DC/DC	6ES7 211-1AD30-0XB0
	CPU 1211C AC/DC/relè	6ES7 211-1BD30-0XB0
	CPU 1211C DC/DC/relè	6ES7 211-1HD30-0XB0
CPU 1212C	CPU 1212C DC/DC/DC	6ES7 212-1AD30-0XB0
	CPU 1212C AC/DC/relè	6ES7 212-1BD30-0XB0
	CPU 1212C DC/DC/relè	6ES7 212-1HD30-0XB0
CPU 1214C	CPU 1214C DC/DC/DC	6ES7 214-1AE30-0XB0
	CPU 1214C AC/DC/relè	6ES7 214-1BE30-0XB0
	CPU 1214C DC/DC/relè	6ES7 214-1HE30-0XB0

Moduli di I/O, moduli di comunicazione e signal board		Numero di ordinazione
Moduli di I/O	SM 1221 8 ingressi 24 V DC	6ES7 221-1BF30-0XB0
	SM 1221 16 ingressi 24 V DC	6ES7 221-1BH30-0XB0
	SM 1222 8 uscite 24 V DC	6ES7 222-1BF30-0XB0
	SM 1222 16 uscite 24 V DC	6ES7 222-1BH30-0XB0
	SM 1222 8 uscite relè	6ES7 222-1HF30-0XB0
	SM 1222 16 uscite relè	6ES7 222-1HH30-0XB0
	SM 1223 8 ingressi 24 V DC / 8 uscite 24 V DC	6ES7 223-1BH30-0XB0
	SM 1223 16 ingressi 24 V DC / 16 uscite 24 V DC	6ES7 223-1BL30-0XB0
	SM 1223 8 ingressi 24 V DC / 8 uscite relè	6ES7 223-1PH30-0XB0
	SM 1223 16 ingressi 24 V DC / 16 uscite relè	6ES7 223-1PL30-0XB0
	SM 1231 4 ingressi analogici	6ES7 231-4HD30-0XB0
	SM 1231 8 ingressi analogici	6ES7 231-4HF30-0XB0
	SM 1232 2 uscite analogiche	6ES7 232-4HB30-0XB0
	SM 1232 4 uscite analogiche	6ES7 232-4HD30-0XB0
	SM 1234 4 ingressi analogici / 2 uscite analogiche	6ES7 234-4HE30-0XB0
Moduli di comunicazione	CM 1241 RS232	6ES7 241-1AH30-0XB0
	CM 1241 RS485	6ES7 241-1CH30-0XB0
Signal board	SB 1223 2 ingressi 24 V DC / 2 uscite 24 V DC	6ES7 223-0BD30-0XB0
	SB 1232 1 uscita analogica	6ES7 232-4HA30-0XB0

Dispositivi HMI	Numero di ordinazione
KTP400 Basic (Mono, PN)	6AV6 647-0AA11-3AX0
KTP600 Basic (Mono, PN)	6AV6 647-0AB11-3AX0
KTP600 Basic (Color, PN)	6AV6 647-0AD11-3AX0

Dispositivi HMI	Numero di ordinazione
KTP1000 Basic (Color, PN)	6AV6 647-0AF11-3AX0
TP1500 Basic (Color, PN)	6AV6 647-0AG11-3AX0

Pacchetto di programmazione	Numero di ordinazione
STEP 7 Basic v10.5	6ES7 822-0AA0-0YA0

Memory card, altri componenti hardware e ingressi analogici		Numero di ordinazione
Memory card	SIMATIC MC 2 MB	6ES7 954-8LB00-0AA0
	SIMATIC MC 24 MB	6ES7 954-8LF00-0AA0
Altri componenti hardware	Alimentatore PSU 1200	6EP1 332-1SH71
	CSM 1277 Switch Ethernet - 4 porte	6GK7 277-1AA00-0AA0
	Simulatore (1214C/1211C - 8 posizioni)	6ES7 274-1XF30-0XA0
	Simulatore (1214C - 14 posizioni)	6ES7 274-1XH30-0XA0
	Prolunga per I/O, 2 m	6ES7 290-6AA30-0XA0
Parti di ricambio	Morsettiera 7 connettori, stagno	6ES7 292-1AG30-0XA0
	Morsettiera 8 connettori, stagno (4/pk)	6ES7 292-1AH30-0XA0
	Morsettiera 11 connettori, stagno (4/pk)	6ES7 292-1AL30-0XA0
	Morsettiera 12 connettori, stagno (4/pk)	6ES7 292-1AM30-0XA0
	Morsettiera 14 connettori, stagno (4/pk)	6ES7 292-1AP30-0XA0
	Morsettiera 20 connettori, stagno (4/pk)	6ES7 292-1AV30-0XA0
	Morsettiera 3 connettori, oro (4/pk)	6ES7 292-1BC0-0XA0
	Morsettiera 6 connettori, oro (4/pk)	6ES7 292-1BF30-0XA0
	Morsettiera 7 connettori, oro (4/pk)	6ES7 292-1BG30-0XA0
	Morsettiera 11 connettori, oro (4/pk)	6ES7 292-1BL30-0XA0

Documentazione	Numero di ordinazione
Sistema di automazione S71200, Manuale di sistema • Tedesco • Inglese • Francese • Spagnolo • Italiano • Cinese	• 6ES7 298-8FA30-8AH0 • 6ES7 298-8FA30-8BH0 • 6ES7 298-8FA30-8CH0 • 6ES7 298-8FA30-8DH0 • 6ES7 298-8FA30-8EH0 • 6ES7 298-8FA30-8KH0
Guida rapida all'S7-1200 • Tedesco • Inglese • Francese • Spagnolo • Italiano • Cinese	• 6ES7 298-8FA30-8AQ0 • 6ES7 298-8FA30-8BQ0 • 6ES7 298-8FA30-8CQ0 • 6ES7 298-8FA30-8DQ0 • 6ES7 298-8FA30-8EQ0 • 6ES7 298-8FA30-8KQ0

Indice analitico

A

- AC
 - Carichi induttivi, 35
- Accesso alla Guida in linea, 16
- Allarmi
 - Descrizione, 42
- Ambienti
 - Industriali, 291
- Ampliamento delle funzionalità dell'S7-1200, 13
- Architettura di interrogazione, 255
- Architettura di interrogazione del master, 255
- Architettura di interrogazione dello slave, 256
- Aree di memoria, 55, 57
- Assegna routine di allarme, 172
- Assistenza, 3
- Assistenza tecnica, 3

B

- Biblioteca del protocollo USS, 183
- Biblioteca globale
 - USS, 183
- Bit di stop, 247
- Blocchi
 - Blocchi dati (DB), 37
 - Blocchi funzionali (FB), 37
 - Blocchi organizzativi (OB), 37, 42
 - Funzioni (FC), 37
- Blocchi di codice, 83
- Blocco
 - Tipi, 37
- Blocco dati
 - Blocco dati di istanza, 55
 - Blocco dati globale, 55, 88
 - Blocco organizzativo (OB), 84
- Blocco dati (DB), 88
- Blocco dati di istanza, 55
- Blocco dati globale, 55, 88
- Blocco di codice
 - DB (blocco dati), 88
 - FB (blocco funzionale), 86
 - FC (funzione), 86
 - Protezione del know-how, 91
- Blocco di gestione dati (DHB), 88
- Blocco di trasmissione (blocco T), 233

Blocco funzionale (FB)

- Blocco dati di istanza, 86
- Parametri di uscita, 87
- Valore iniziale, 86

Blocco organizzativo

- Classi di priorità, 42
- Configurazione dell'esecuzione, 86
- Creazione, 85
- Di ciclo, multipli, 85
- Elaborazione, 84
- Funzione, 42
- Richiamo, 42

Break, 250, 251

Budget di potenza, 22, 333
esempio, 334, 335

Buffer di diagnostica, 51, 283

C

- Carattere di fine messaggio, 253
- Carattere inizio messaggio, 251
- Caricamento nel dispositivo, 228
- Carichi delle lampade, 36
- Carichi induttivi, 35
- Classe di priorità
 - Descrizione, 42
- Classe di sicurezza, 293
- CM 1241 RS232, dati tecnici, 330
- CM 1241 RS485, dati tecnici, 329
- Codici di stato USS, 193
- Collegamento di rete, 77
 - Diverse CPU, 232
- Combinazione logica di bit, 95
- Compatibilità elettromagnetica (CEM), 291
- Comunicazione
 - Architettura di interrogazione, 255
 - Biblioteche, 245
 - Carico, 49
 - Collegamento hardware, 219
 - Controllo del flusso, 247
 - Indirizzo IP, 78, 225
 - Parametri di invio e ricezione, 249
 - Rete, 219
- Comunicazione di rete, 219
- Comunicazione Ethernet, 217
- Comunicazione PtP, 245
- Comunicazione punto a punto, 245
- Comunicazione seriale, 245

- Comunicazione TCP/IP, 217
- Condizioni ambientali, 292
- Condizioni di fine, 252
- Condizioni di inizio, 250
- Configurazione
 - Comunicazione da PLC a PLC, 231
 - Da HMI a CPU, 230
 - HSC (contatore veloce), 113
 - Indirizzo IP, 78, 225
 - Interfacce di comunicazione, 246
 - Parametri di avvio, 38, 65
 - Porta Industrial Ethernet, 78, 225
 - Porte, 246
 - PROFINET, 78, 225
 - Ricezione dei messaggi, 250
 - Rilevamento, 73
 - Tempo di ciclo, 49
- Configurazione dei dispositivi, 71, 220
 - Aggiungi nuovo dispositivo, 72
 - Collegamento di rete, 77
 - Configurazione dei moduli, 76
 - Configurazione della CPU, 74
 - Inserimento dei moduli, 75
 - Porta Ethernet, 78, 225
 - PROFINET, 78, 225
 - Rilevamento, 73
- Configurazione dei messaggi
 - Istruzioni, 255
 - Ricezione, 250
 - Trasmissione, 249
- Configurazione dei parametri
 - Ricezione, 238
 - Trasmissione, 234
- Configurazione dei parametri di invio, 234
- Configurazione dei parametri di ricezione, 238
- Configurazione della porta, 246
 - Istruzioni, 255
- Configurazione della ricezione dei messaggi, 250
- Configurazione della trasmissione dei messaggi, 249
- Configurazione dell'istruzione TRCV_C, 238
- Configurazione dell'istruzione TSEND_C, 234
- Configurazione hardware, 71
 - Aggiungi nuovo dispositivo, 72
 - Collegamento di rete, 77
 - Configurazione dei moduli, 76
 - Configurazione della CPU, 74
 - Inserimento dei moduli, 75
 - Porta Ethernet, 78, 225
 - PROFINET, 78, 225
 - Rilevamento, 73
- Connettore
 - Montaggio e smontaggio, 32
- Connettore di bus, 14
- Contatore
 - veloce (HSC), 111
 - veloce (HSC): Configurazione, 113
- Contatore veloci, 111
- Controllo del flusso, 247
 - Configurazione, 247
- Controllo del flusso hardware, 247
- Controllo del flusso software, 248
- Controllo del programma, 93
- Controllo del tempo di ciclo, CPU online, 283
- Controllo dell'utilizzo della memoria, CPU online, 283
- CPU
 - 1211C, dati tecnici, 294
 - 1211C, schemi elettrici, 298
 - 1212C, dati tecnici, 300
 - 1212C, schemi elettrici, 304
 - 1214C, dati tecnici, 305
 - 1214C, schemi elettrici, 309
 - Aggiungi nuovo dispositivo, 72
 - Attivazione di un collegamento online, 281
 - Budget di potenza, 22
 - Caricamento nel dispositivo, 228
 - Carichi delle lampade, 36
 - Carichi induttivi, 35
 - Collegamento di rete, 77
 - Configurazione dei dispositivi, 71
 - Configurazione della comunicazione con l'HMI, 229
 - Configurazione di diverse, 231
 - CPU non specificata, 73
 - Creazione di una scheda di programma, 68
 - Creazione di una scheda di trasferimento, 66
 - Descrizione, 11
 - Elaborazione dell'avviamento, 40
 - Esecuzione del programma, 38
 - Indirizzo IP, 78, 225
 - Indirizzo MAC, 241
 - Inserimento dei moduli, 75
 - Istruzioni per il cablaggio, 32, 34
 - Istruzioni per l'isolamento, 33
 - Livelli di sicurezza, 54
 - Messa a terra, 34
 - Modi di funzionamento, 39
 - Modo STOP, 286
 - Online, 282
 - Pannello operatore per CPU online, 282
 - Parametri di avvio, 38, 65
 - Parametri di configurazione, 74
 - Password persa, 55
 - Porta Ethernet, 78, 225
 - Procedimenti di montaggio, 26
 - PROFINET, 78, 225

Protezione mediante password, 54
 Requisiti di alimentazione, 333
 Ripristino in caso di perdita della password, 55
 Scheda di programma, 68
 Scheda di trasferimento, 66
 Scheda di trasferimento vuota, 55
 Signal board (SB), 13
 Tabella comparata, 12
 Tabelle di controllo, 284
 Tempo di ciclo, 49
 Zona termica, 24
 CPU non specificata, 73
 CPU online, 282
 Controllo del tempo di ciclo, 283
 Controllo dell'utilizzo della memoria, 283
 Pannello operatore, 282
 Creazione di una connessione di rete, 77
 CTS, 247

D

Dati tecnici, 289
 Ambienti, 291
 Compatibilità elettromagnetica (CEM), 291
 Condizioni ambientali, 292
 CPU 1211C, 294
 CPU 1212C, 300
 CPU 1214C, 305
 Durata di servizio dei relè, 294
 Generali, 289
 Memory card, 330
 Moduli di I/O analogici, 318
 Modulo di comunicazione CM 1241 RS232, 330
 Modulo di comunicazione CM 1241 RS485, 329
 Modulo di I/O SM 1221, 311
 Modulo di I/O SM 1222, 313
 Modulo di I/O SM 1223, 315
 Omologazione ATEX, 290
 Omologazione CE, 289
 Omologazione C-Tick, 291
 Omologazione cULus, 290
 Omologazione FM, 290
 Omologazione nel settore marittimo, 291
 Protezione, 293
 SB 1223, 325
 SB 1223, 327
 Schemi elettrici: SM 1231, 1232, 1234, 323
 Signal board digitali (SB), 325
 simulatori di ingressi, 331
 SM 1221, schemi elettrici, 312
 SM 1222, schemi elettrici, 314
 SM 1223, schemi elettrici, 317

Tensione dei moduli di I/O analogici, 320
 Tensioni nominali, 293
 Dati tecnici dei moduli di I/O analogici, 318
 Dati tecnici delle memory card, 330
 Dati tecnici delle signal board digitali (SB), 325
 Dati tecnici dell'SB 1223, 325, 327
 Dati tecnici generali, 289
 DB (blocco dati), 88
 DC
 Carichi induttivi, 35
 Descrizioni comando, 16
 Disancoraggio della Guida in linea, 17
 Dispositivi HMI
 Collegamento di rete, 77
 Descrizione, 20
 Documentazione, 16
 Durata di servizio dei relè, 294

E

EN e ENO (flusso di corrente), 90
 Errori
 Errori di diagnostica, 47
 Errori temporali, 46
 Istruzioni PtP, 275
 Errori di configurazione della ricezione, 276
 Errori di configurazione della trasmissione, 276
 Errori di configurazione delle porte, 275
 Errori di gestione del segnale, 276
 Errori di runtime di trasmissione, 277
 Errori vari relativi ai parametri PtP, 278
 Esecuzione degli eventi, 44
 Esecuzione del programma, 37
 Ethernet
 Collegamento di rete, 77
 Indirizzo IP, 78, 225

F

FB (blocco funzionale), 86
 FC (funzione), 86
 Fine messaggio, 252
 Funzione (FC), 86
 Funzione matematica in virgola mobile, 121
 Funzioni matematiche, 116
 FUP (schema logico), 90

G

Gap intercaratteri, 253
 Guida, 16

- Disancoraggio, 17
- Ingrandimento, 17
- Stampa, 19
- Visualizzazione degli argomenti e del sommario, 17
- Guida alle operazioni di base
 - Descrizioni comando, 16
 - Descrizioni comando con casella a discesa, 16
 - Documentazione, 16
 - Guida in linea, 16
 - Guida rapida, 16
 - Roll-out, 16
 - Sistema di informazione, 16
 - Vista portale e vista progetto, 15
- Guida DIN, 25
- Guida in linea, 16
 - Disancoraggio, 17
 - Ingrandimento della finestra della Guida, 17
 - Stampa, 19
 - Visualizzazione degli argomenti e del sommario, 17
- Guida rapida, 16

H

- HMI
 - Configurazione della comunicazione PROFINET, 229
- Hotline, 3
- HSC (contatore veloce), 111
 - Configurazione, 113

I

- I/O
 - Carichi induttivi, 35
 - Indicatori di stato analogici, 280
 - Indicatori di stato digitali, 280
 - Indirizzamento, 59
- Indicatori di stato degli I/O analogici, 280
- Indicatori di stato degli I/O digitali, 280
- Indicatori LED, 256, 279
- Indirizzo IP, 78, 79, 225, 226
 - Assegnazione, 220, 227
 - Assegnazione online, 223
 - Configurazione, 78, 225
- Indirizzo IP del router, 79, 226
- Indirizzo IP, impostazione nella CPU online, 282
- Indirizzo MAC, 78, 225, 241
- Informazioni sui contatti, 3
- Ingrandimento della finestra della Guida in linea, 17
- Inizio messaggio, 250
- Inserimento di un dispositivo

- CPU non specificata, 73
- Inserimento nella coda d'attesa, 44
- Interfacce di comunicazione
 - Configurazione, 246
 - Programmazione, 254
- Interfaccia PROFINET
 - Proprietà dell'indirizzo Ethernet, 79, 226
 - Proprietà per la sincronizzazione dell'ora, 243
- Istanza singola
 - Esempio, 87
- Istruzione AND, 131
- Istruzione Arrotonda numero, 126
- Istruzione Codifica (ENCO), 131
- Istruzione Compila campo senza interruzione (UFILL_BLK), 124
- Istruzione Contatore veloce (HSC), 108
- Istruzione Copia area (MOVE_BLK), 122
- Istruzione Copia campo senza interruzione (UMOVE_BLK), 122
- Istruzione Crea complemento a uno (INV), 131
- Istruzione CTRL_PWM, 180
- Istruzione DEC (decrementa), 118
- Istruzione Decodifica (DECO), 131
- Istruzione di allarme di ritardo CAN_DINT, 176
- Istruzione di allarme di ritardo SRT_DINT, 176
- Istruzione di fronte di discesa, 100
- Istruzione di fronte di salita, 100
- Istruzione di trasferimento, 122
- Istruzione DIS_AIRT di elaborazione allarmi, 177
- Istruzione Dividi (DIV), 116
- Istruzione EN_AIRT di elaborazione allarmi, 177
- Istruzione Etichetta di salto, 129
- Istruzione Fai ruotare verso destra (ROR), 135
- Istruzione Fai ruotare verso sinistra (ROL), 135
- Istruzione Genera da un numero in virgola mobile il numero intero inferiore (FLOOR), 127
- Istruzione Genera da un numero in virgola mobile il numero intero superiore (CEIL), 127
- Istruzione Genera numero intero (TRUNC), 126
- Istruzione Genera valore assoluto (ABS), 119
- Istruzione Imposta, 98
- Istruzione Imposta valore limite, 120
- Istruzione INC (incrementa), 118
- Istruzione Inserire i dati nel campo (FILL_BLK), 124
- Istruzione JMPN, 129
- Istruzione MAX (massimo), 119
- Istruzione MIN (minimo), 119
- Istruzione MOD (modulo), 117
- Istruzione Modifica disposizione, 125
- Istruzione Multiplexaggio (MUX), 133
- Istruzione NEG (negazione), 117
- Istruzione Normazione (NORM), 128

- Istruzione OR, 131
Istruzione PID_Compact, 178
Istruzione PORT_CFG (Configurazione della porta), 258
Istruzione RCV_CFG (Configura ricezione), 261
Istruzione RCV_PTP (Ricevi punto a punto), 271
Istruzione RCV_RST (Resetta buffer di ricezione), 272
Istruzione RE_TRIGR, 154
Istruzione Resetta, 98
Istruzione Resetta il timer (RT), 102
Istruzione Riporta in scala (SCALE_X), 128
Istruzione RT (Resetta il timer), 102
Istruzione S_CONV, 141
Istruzione Salta indietro (RET), 130
Istruzione Salto nel blocco se 1 (condizionato) (JMP), 129
Istruzione Seleziona (SEL), 133
Istruzione SEND_CFG (Configura trasmissione), 260
Istruzione SEND_PtP (Trasmetti dati punto a punto), 268
Istruzione Separa evento, 172
Istruzione SGN_GET (Leggi segnali RS232), 273
Istruzione SGN_SET (Imposta segnali RS232), 274
Istruzione Somma (ADD), 116
Istruzione Sposta verso destra (SHR), 134
Istruzione Sposta verso sinistra (SHL), 134
Istruzione STP (Arresta ciclo di scansione del PLC), 155
Istruzione STRG_VAL, 141
Istruzione T_ADD, 136
Istruzione T_CONV, 136
Istruzione T_DIFF, 136
Istruzione T_SUB, 136
Istruzione TCON, 164
Istruzione TDISCON, 164
Istruzione Temporizzatore come impulso prolungato (TP), 102
Istruzione Temporizzatore come ritardo all'inserzione (TON), 102
Istruzione Temporizzatore come ritardo alla disinserzione (TOF), 102
Istruzione Temporizzatore come ritardo all'inserzione con memoria (TONR), 102
Istruzione TRCV, 164
Istruzione TRCV_C, 159, 237
Istruzione TSEND, 164
Istruzione TSEND_C, 159, 233
Istruzione USS_DRV, 186
Istruzione USS_PORT, 189
Istruzione USS_RPM, 190
Istruzione USS_WPM, 191
Istruzione VAL_STRG, 141
Istruzione Valore compreso nel campo, 115
Istruzione Valore fuori campo, 115
Istruzione Verifica nullità, 115
Istruzione Verifica validità, 115
Istruzione XOR (OR esclusivo), 131
Istruzioni
Allarme: ATTACH, 172
Allarme: CAN_DINT, 176
Allarme: DETACH, 172
Allarme: DIS_AIRT, 177
Allarme: EN_AIRT, 177
Allarme: SRT_DINT, 176
AND, 131
Arrotonda numero, 126
Calendario, 136
Carichi delle lampade, 36
Carichi induttivi, 35
Codici di stato USS, 193
Codifica (ENCO), 131
Combinazione logica di bit, 95
Compila campo senza interruzione (UFILL_BLK), 124
Confronto, 114
Contatore, 106
Contatore veloce (HSC), 108
Conversione, 126
Conversione da stringa in valore: S_CONV, 141
Conversione da stringa in valore: STRG_VAL, 141
Conversione da valore in stringa: S_CONV, 141
Conversione da valore in stringa: VAL_STRG, 141
Copia area (MOVE_BLK), 122
Copia campo senza interruzione (UMOVE_BLK), 122
Crea complemento a uno (INV), 131
CTRL_PWM, 180
Data, 136
DEC (decrementa), 118
Decodifica (DECO), 131
Dividi (DIV), 116
Entro il campo, 115
Etichetta di salto, 129
Fai ruotare verso destra (ROR), 135
Fai ruotare verso sinistra (ROL), 135
Fronte di discesa, 100
Fronte di salita, 100
Fuori campo, 115
Genera da un numero in virgola mobile il numero intero inferiore (FLOOR), 127
Genera da un numero in virgola mobile il numero intero superiore, 127
Genera numero intero (TRUNC), 126
Genera valore assoluto (ABS), 119

- GET_ERROR, 159
- Imposta, 98
- Imposta valore limite, 120
- INC (incrementa), 118
- Inserire i dati nel campo (FILL_BLK), 124
- Isolamento, 33
- Istruzioni per il cablaggio, 32, 34
- MAX (massimo), 119
- Messa a terra, 34
- MIN (minimo), 119
- MOD (modulo), 117
- Modifica disposizione, 125
- Moltiplica (MUL), 116
- Montaggio, 21
- Multiplexaggio (MUX), 133
- NEG (negazione), 117
- Normazione (NORM), 128
- Operazioni di trasferimento, 122
- Operazioni in virgola mobile, 121
- OR, 131
- Orologio, 138
- Orologio: Leggi data e ora di sistema (RD_SYS_T), 138
- Orologio: Leggi data e ora locale (RD_LOC_T), 138
- Orologio: Scrivi data e ora di sistema (WR_SYS_T), 138
- PID_Compact, 178
- PORT_CFG (Configurazione della porta), 258
- Procedimenti di montaggio, 25
- RCV_CFG (Configura ricezione), 261
- RCV_PtP (Ricevi punto a punto), 271
- RCV_RST (Resetta buffer di ricezione), 272
- RE_TRIGR, 48, 154
- Resetta, 98
- Riporta in scala (SCALE_X), 128
- Salta indietro (RET), 130
- Salto nel blocco se 1 (condizionato) (JMP), 129
- Seleziona (SEL), 133
- SEND_CFG (Configura trasmissione), 260
- SEND_PtP (Trasmetti dati punto a punto), 268
- SGN_GET (Leggi segnali RS232), 273
- SGN_SET (Imposta segnali RS232), 274
- Somma (ADD), 116
- Sottrai (SUB), 116
- Sposta verso destra (SHR), 134
- Sposta verso sinistra (SHL), 134
- STP (Arresta ciclo di scansione del PLC), 155
- T_ADD, 136
- T_CONV, 136
- T_DIFF, 136
- T_SUB, 136
- TCON, 164
- TDISCON, 164
- Tempo, 136
- Temporizzatore, 102
- Temporizzatore: RT (Resetta il timer), 102
- Temporizzatore: TOF (Temporizzatore come ritardo alla disinserzione), 102
- Temporizzatore: TON (Temporizzatore come ritardo all'inserzione), 102
- Temporizzatore: TONR (Temporizzatore come ritardo all'inserzione con memoria), 102
- Temporizzatore: TP (Temporizzatore come impulso), 102
- TRCV, 164
- TRCV_C, 159, 237
- TSEND, 164
- TSEND_C, 159, 233
- USS_DRV, 186
- USS_PORT, 189
- USS_RPM, 190
- USS_WPM, 191
- Verifica nullità, 115
- Verifica validità, 115
- XOR (OR esclusivo), 131
- Istruzioni di confronto, 114
- Istruzioni di conteggio, 106
- Istruzioni di conversione, 126
- Istruzioni di conversione da stringa in valore, 141
- Istruzioni di conversione da valore in stringa, 141
- Istruzioni di orologio, 138
 - Leggi data e ora di sistema (RD_SYS_T), 138
 - Leggi data e ora locale (RD_LOC_T), 138
 - Scrivi data e ora di sistema (WR_SYS_T), 138
- Istruzioni di temporizzazione, 102
- Istruzioni Ethernet
 - TCON, 164
 - TDISCON, 164
 - TRCV, 164
 - TRCV_C, 159
 - TSEND, 164
 - TSEND_C, 159
- Istruzioni Moltiplica (MUL), 116
- Istruzioni per il cablaggio
 - Messa a terra, 34
 - Presupposti, 32
- Istruzioni per il calendario, 136
- Istruzioni per la data, 136
- Istruzioni per la data e l'ora, 136
- Istruzioni per l'isolamento, 33
- Istruzioni Sottrai (SUB), 116

K

KOP (schema a contatti), 89

L

Latenza degli allarmi, 46

Linea inattiva, 250, 251

Livello di protezione

Blocco di codice, 91

CPU, 54

Password persa, 55

Lunghezza

Messaggio, 253

Lunghezza del messaggio, 253

Lunghezza m, 253

Lunghezza massima del messaggio, 253

Lunghezza n, 253

M

Maschera della sottorete, 78, 225

MB_COMM_LOAD, 194

MB_MASTER, 196

MB_SLAVE, 206

Memoria

I (immagine di processo degli ingressi), 57

L (memoria locale), 55

M (merker), 58

Memoria a ritenzione, 49

Memoria di caricamento, 49

Memoria di lavoro, 49

Memoria temporanea, 59

Merker di clock, 52

Merker di sistema, 52

Q (immagine di processo delle uscite), 57

Memory card

Configurazione dei parametri di avvio, 65

Password persa, 55

Scheda di programma, 68

Scheda di trasferimento, 66

Scheda di trasferimento vuota, 55

MODBUS, 194

MB_Master, 196

MB_SLAVE, 206

Modo RUN, 39, 41

Modo STOP, 39, 286

Moduli

Modulo di comunicazione (CM), 14

Modulo di I/O (SM), 14

Parametri di configurazione, 76

Signal board (SB), 13

Tabella comparata, 13

Zona termica, 24

Moduli di comunicazione

RS232 e RS485, 245

Moduli di comunicazione RS232 e RS485, 245

Moduli di I/O

SM 1221, dati tecnici, 311

SM 1222, dati tecnici, 313

SM 1223, dati tecnici, 315

Tabelle di controllo, 284

Modulo di comunicazione

Aggiungi nuovo dispositivo, 72

Configurazione dei dispositivi, 71

Inserimento dei moduli, 75

Modulo di comunicazione (CM), 256

dati tecnici, 329

Descrizione, 14

Montaggio, 29

Requisiti di alimentazione, 333

Ricezione dei dati, 271

Smontaggio, 29

Tabella comparata, 13

Modulo di comunicazione (CM), biblioteca USS, 183

Modulo di I/O (SM)

Aggiungi nuovo dispositivo, 72

Configurazione dei dispositivi, 71

Descrizione, 14

Inserimento dei moduli, 75

Montaggio, 28

Requisiti di alimentazione, 333

Smontaggio, 28

Tabella comparata, 13

Montaggio

Budget di potenza, 22

Carichi delle lampade, 36

Carichi induttivi, 35

CPU, 26

Descrizione, 21, 25

Dimensioni, 24

Isolamento, 33

Istruzioni, 21

Istruzioni per il cablaggio, 32, 34

Istruzioni per l'isolamento, 33

Messa a terra, 34

Modulo di comunicazione (CM), 29

Modulo di I/O (SM), 28

Morsettiera, 32

Portale TIA, 15

Quote di montaggio, 24

Signal board (SB), 31

Spazio libero, 22

STEP 7, 15

- Zona termica, 24
- Morsettiera
 - Montaggio, 32

N

- Network Time Protocol (NTP), 243
- Numeri
 - Reali, 61
 - Virgola mobile, 61

O

- Omologazione ATEX, 290
- Omologazione CE, 289
- Omologazione C-Tick, 291
- Omologazione cULus, 290
- Omologazione FM, 290
- Omologazione nel settore marittimo, 291
- Online, attivazione di un collegamento, 281
- Ora, impostazione nella CPU online, 282
- Orologio
 - Orologio hardware, 51

P

- Parametri di avvio, 38, 65
- Parametri di configurazione
 - CPU, 74
 - Moduli, 76
 - Porta Ethernet, 78, 225
 - PROFINET, 78, 225
- Parametri di uscita, 87
- Parametrizzazione, 87
- Parità, 247
- Password, 55
- Password persa, 55
- PLC
 - Descrizione, 11
 - Progettazione del sistema, 81
 - Utilizzo dei blocchi, 82
- Portale TIA
 - Aggiungi nuovo dispositivo, 72
 - Collegamento di rete, 77
 - Configurazione dei dispositivi, 71
 - Configurazione dei moduli, 76
 - Configurazione della CPU, 74
 - Inserimento dei moduli, 75
 - Montaggio, 15
 - Porta Ethernet, 78, 225
 - PROFINET, 78, 225

- Vista portale, 15
- Vista progetto, 15
- Posizione del carattere
 - Lunghezza del messaggio, 253
- Priorità di elaborazione, 44
- PROFINET, 217
 - Collegamento di rete, 77
 - Indirizzo IP, 78, 225
 - Test di una rete, 227
- Progettazione di un sistema PLC, 81, 82
- Progetto
 - Limitazione dell'accesso alla CPU, 54
 - Password persa, 55
 - Protezione di un blocco di codice, 91
 - Scheda di programma, 68
 - Scheda di trasferimento, 66
 - Scheda di trasferimento vuota, 55
- Programmazione
 - CPU non specificata, 73
 - Flusso di corrente (EN e ENO), 90
 - FUP (schema logico), 90
 - Istruzioni PtP, 254
 - KOP (contatti), 89
 - Lineare, 82
 - Strutturata, 82
- Programmazione lineare, 82
- Programmazione PtP, 254
- Programmazione punto a punto, 254
- Programmazione strutturata, 82, 83
- Protezione dalla copia, 91
- Protezione del know-how, 91
- Protezione mediante password
 - Blocco di codice, 91
 - CPU, 54
 - Password persa, 55
 - Scheda di trasferimento vuota, 55
- Protocollo
 - Comunicazione, 245
 - Freeport, 245
- Protocollo freeport, 245
- PTO (uscita di treni di impulsi), 180
- PWM
 - Istruzione CTRL_PWM, 180

R

- Requisiti di alimentazione
 - calcolo, 334, 335
- Richiamo di blocco
 - Concetti base, 37
 - Richiamo di un'istanza singola o di una multiistanza, 87

Rilevamento, 73
 Ritardo RTS OFF, 249
 Ritardo RTS ON, 249
 Roll-out, 16
 Router IP, 78, 225
 RTS, 247
 RTS disattivato, 247
 RTS sempre ON, 248

S

S7-1200

Aggiungi nuovo dispositivo, 72
 Ampliamento delle funzionalità, 13
 Budget di potenza, 22
 Carichi delle lampade, 36
 Carichi induttivi, 35
 Collegamento di rete, 77
 Configurazione dei dispositivi, 71
 Configurazione dei moduli, 76
 Configurazione dei parametri della CPU, 74
 CPU, 11
 Descrizione del montaggio, 25
 Dispositivi HMI, 20
 Indirizzo IP, 78, 225
 Inserimento dei moduli, 75
 Istruzioni per il cablaggio, 32, 34
 Istruzioni per l'isolamento, 33
 Messa a terra, 34
 Modulo di comunicazione (CM), 14
 Modulo di I/O (SM), 14
 Montaggio di un CM, 29
 Montaggio di un SB, 31
 Montaggio di un SM, 28
 Morsettiera, 32
 Parametri di avvio, 38, 65
 Password persa, 55
 Porta Ethernet, 78, 225
 Procedimenti di montaggio della CPU, 26
 PROFINET, 78, 225
 Protezione mediante password, 54
 Quote di montaggio, 24
 Scheda di programma, 68
 Scheda di trasferimento, 66
 Scheda di trasferimento vuota, 55
 Signal board (SB), 13
 Spazio libero, 22
 Tabella comparata dei modelli di CPU, 12
 Tempo di ciclo, 49
 Zona termica, 24
 SB 1223, schema elettrico, 326
 SB 1232, schema elettrico, 328

Scheda di programma, 68
 Configurazione dei parametri di avvio, 65
 Scheda di trasferimento, 66
 Configurazione dei parametri di avvio, 65
 Password persa, 55
 Scheda di trasferimento vuota, 55
 Schemi elettrici
 CPU 1211C, 298
 CPU 1212C, 304
 CPU 1214C, 309
 Modulo di I/O SM 1221, 312
 Modulo di I/O SM 1222, 314
 Modulo di I/O SM 1223, 317
 SB 1223, 326
 SB 1232, 328
 SM 1231, 1232, 1234, 323
 Sequenza di caratteri
 Fine messaggio, 253
 Inizio messaggio, 251
 Servizio clienti, 3
 Sicurezza
 Blocco di codice, 91
 CPU, 54
 Password persa, 55
 Signal board (SB)
 Configurazione dei dispositivi, 71
 Descrizione, 13
 Inserimento dei moduli, 75
 Montaggio, 31
 Requisiti di alimentazione, 333
 Smontaggio, 31
 Tabella comparata, 13
 Signal board (SM)
 Aggiungi nuovo dispositivo, 72
 Simulatori di ingressi, 331
 Sistema di informazione, 16
 Disancoraggio, 17
 Ingrandimento, 17
 Stampa, 19
 Visualizzazione degli argomenti e del sommario, 17
 Spazio libero per l'installazione, 22
 Stampa degli argomenti della Guida, 19
 STEP 7
 Aggiungi nuovo dispositivo, 72
 Collegamento di rete, 77
 Configurazione dei dispositivi, 71
 Configurazione dei moduli, 76
 Configurazione della CPU, 74
 Inserimento dei moduli, 75
 Montaggio, 15
 Porta Ethernet, 78, 225
 PROFINET, 78, 225

Vista portale, 15
Vista progetto, 15
Struttura del programma, 83

T

Tabella comparata
Dispositivi HMI, 20
Modelli di CPU, 12
Tabella comparata dei moduli, 13
Tabelle di controllo, 93, 284
Tempo di attesa, 247
Tempo di ciclo, 48, 49
Tensione dei moduli di I/O analogici, 320
Tensioni nominali, 293
Test del programma, 93
Tipi di dati, 60
Array, 62
DTL, 62
STRING, 61
Tipo di dati Data and Time Long (DTL), 62
Tipo di dati DTL, 62
Tipo di dati String, 61
TOF, istruzione Temporizzatore come ritardo alla disinserzione, 102
TON, istruzione Temporizzatore come ritardo all'inserzione, 102
TONR, istruzione Temporizzatore come ritardo all'inserzione con memoria, 102
TP, istruzione Temporizzatore come impulso prolungato, 102
Transizione da RUN a STOP, 53
Trasmissione dei dati, avvio, 268
TSAP (transport service access points), 235, 239

U

Uscita di treni di impulsi (PTO), 180

V

Valori di ritorno durante il runtime di ricezione, 277
Valori restituiti
Istruzioni PtP, 275
Valori restituiti dalle istruzioni PtP, 275
Velocità di trasmissione, 247
Vista portale, 15
Aggiungi nuovo dispositivo, 72
Configurazione dei moduli, 76
Configurazione della CPU, 74
Configurazione della porta Ethernet, 78, 225

Inserimento dei moduli, 75
PROFINET, 78, 225
Vista progetto, 15
Aggiungi nuovo dispositivo, 72
Collegamento di rete, 77
Configurazione dei dispositivi, 71
Configurazione dei moduli, 76
Configurazione dei parametri della CPU, 74
Configurazione della porta Ethernet, 78, 225
Inserimento dei moduli, 75
PROFINET, 78, 225

Visualizzazione degli argomenti e del sommario (Guida in linea), 17

W

Watchdog, 154

X

XON / XOFF, 248

Z

Zona termica, 24

