

Modicon M258 Logic Controller

Funzioni di sistema e variabili

Guida della libreria Modicom M258

PLCSystem

06/2011

Questa documentazione contiene la descrizione generale e/o le caratteristiche tecniche dei prodotti qui contenuti. Questa documentazione non è destinata e non deve essere utilizzata per determinare l'adeguatezza o l'affidabilità di questi prodotti relativamente alle specifiche applicazioni dell'utente. Ogni utente o specialista di integrazione deve condurre le proprie analisi complete e appropriate del rischio, effettuare la valutazione e il test dei prodotti in relazioni all'uso o all'applicazione specifica. Né Schneider Electric né qualunque associata o filiale deve essere tenuta responsabile o perseguibile per il cattivo uso delle informazioni ivi contenute. Gli utenti possono inviarci commenti e suggerimenti per migliorare o correggere questa pubblicazione.

È vietata la riproduzione totale o parziale del presente documento in qualunque forma o con qualunque mezzo, elettronico o meccanico, inclusa la fotocopiatura, senza esplicito consenso scritto di Schneider Electric.

Durante l'installazione e l'uso di questo prodotto è necessario rispettare tutte le normative locali, nazionali o internazionali in materia di sicurezza. Per motivi di sicurezza e per assicurare la conformità ai dati di sistema documentati, la riparazione dei componenti deve essere effettuata solo dal costruttore.

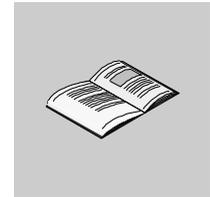
Quando i dispositivi sono utilizzati per applicazioni con requisiti tecnici di sicurezza, occorre seguire le istruzioni più rilevanti.

Un utilizzo non corretto del software Schneider Electric (o di altro software approvato) con prodotti hardware Schneider Electric può costituire un rischio per l'incolumità personale o provocare danni alle apparecchiature.

La mancata osservanza di queste indicazioni può costituire un rischio per l'incolumità personale o provocare danni alle apparecchiature.

© 2011 Schneider Electric. Tutti i diritti riservati.

Indice

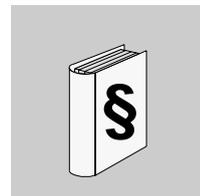


	Informazioni di sicurezza	7
	Informazioni su...	9
Capitolo 1	Variabili di sistema del M258	11
1.1	Variabili di sistema: definizione e uso	12
	Variabili di sistema	13
	Uso delle variabili di sistema	15
1.2	Strutture PLC_R e PLC_W	17
	PLC_R: variabili di sistema in sola lettura del controller	18
	PLC_W: variabili di sistema in lettura/scrittura del controller	22
1.3	Strutture SERIAL_R e SERIAL_W	23
	SERIAL_R[0..1]: variabili di sistema in sola lettura della linea seriale ...	24
	SERIAL_W[0..1]: variabili di sistema in lettura/scrittura della linea seriale	25
1.4	Strutture ETH_R e ETH_W	26
	ETH_R: variabili di sistema in sola lettura porta Ethernet	27
	ETH_W: variabili di sistema lettura/scrittura porta Ethernet	31
1.5	Struttura TM5_MODULE_R	32
	TM5_MODULE_R[1..254]: Variabili di sistema di sola lettura dei moduli TM5	32
Capitolo 2	Funzioni di sistema del M258	33
2.1	Funzioni di lettura M258	34
	DM72FGetImmediateInput: lettura dell'ingresso di un I/O Expert integrato	35
	getTM5Delay: Numero di cicli del bus TM5 senza scambio valido	36
	IsFirstMastColdCycle: indica se il ciclo è il primo ciclo MAST dopo un avvio a freddo	38
	IsFirstMastCycle: indica se il ciclo è il primo ciclo MAST	39
	IsFirstMastWarmCycle: indica se il ciclo è il primo ciclo MAST dopo un avvio a caldo	41
2.2	Funzioni di scrittura del M258	42
	SetLEDBehaviour: determina il comportamento di un LED	43
	SetRTCDrift: regolazione settimanale dell'orologio in tempo reale	45

Capitolo 3	Libreria M258 PLCSystem - Tipi di dati	47
3.1	Tipi di dati delle variabili di sistema PLC_R/W	48
	PLC_R_APPLICATION_ERROR: codici di stato degli errori applicazione rilevati	49
	PLC_R_BOOT_PROJECT_STATUS: codici di stato del progetto di avvio	50
	PLC_R_IO_STATUS: Codici di stato degli I/O	51
	PLC_R_STATUS: Codici di stato del controller	52
	PLC_R_STOP_CAUSE: codici causa transizione da RUN a un altro stato	53
	PLC_R_TERMINAL_PORT_STATUS: codici dello stato di connessione della porta di programmazione	54
	PLC_R_USB_HOST_STATUS: codici dello stato di connessione della porta host USB	55
	PLC_W_COMMAND: codici dei comandi di controllo	56
3.2	Tipi di dati delle variabili di sistema ETH_R/W	57
	ETH_R_IP_MODE: codici sorgente indirizzo IP	58
	ETH_R_FRAME_PROTOCOL: codici del protocollo di trasmissione dei frame	59
	ETH_R_PORT_DUPLEX_STATUS: codici della modalità di trasmissione	60
	ETH_R_PORT_LINK_STATUS: codici relativi alla direzione del collegamento di comunicazione	61
	ETH_R_PORT_SPEED: codici relativi alla velocità di comunicazione della porta Ethernet	62
	ETH_R_PORT_IP_STATUS: Codici di stato porta Ethernet TCP/IP	63
	ETH_R_RUN_IDLE: Codici di stato di inattività e funzionamento di Ethernet/IP	64
3.3	Tipi di dati delle variabili di sistema TM5_MODULE_R/W	65
	TM5_MODULE_STATE: codici di stato dei moduli di espansione TM5	65
3.4	Tipi di dati delle funzioni di sistema	66
	LED_ID: SetLEDBehaviour Funzione LedId - Codici dei parametri	67
	LED_BHV: SetLEDBehaviour Funzione LedBhv - Codici dei parametri	68
	LED_BHV_ERROR: Codici errori rilevati della funzione SetLEDBehaviour	69
	LED_COLOR: SetLEDBehaviour Funzione LedColor - Codici dei parametri	70
	RTCSETDRIFT_ERROR: Codici errori rilevati della funzione SetRTCDrift	71
	DAY_OF_WEEK: Codici dei parametri giorno della funzione SetRTCDrift	72
	ORA: Tipo di parametro ora della funzione SetRTCDrift	73
	MINUTO: Tipo di parametro minuto della funzione SetRTCDrift	74
Appendici		75

Appendice A	Rappresentazione di funzioni e blocchi funzione	77
	Differenze tra una funzione e un blocco funzione	78
	Come utilizzare una funzione o un blocco funzione in linguaggio IL	79
	Come utilizzare una funzione o un blocco funzione in linguaggio ST	82
Glossario	85
Indice analitico	93

Informazioni di sicurezza



Informazioni importanti

AVVISO

Leggere attentamente queste istruzioni e osservare l'apparecchiatura per familiarizzare con i suoi componenti prima di procedere ad attività di installazione, uso o manutenzione. I seguenti messaggi speciali possono comparire in diverse parti della documentazione oppure sull'apparecchiatura per segnalare rischi o per richiamare l'attenzione su informazioni che chiariscono o semplificano una procedura.



L'aggiunta di questo simbolo a un'etichetta di Pericolo o Avvertenza relativa alla sicurezza indica che esiste un rischio da shock elettrico che può causare lesioni personali se non vengono rispettate le istruzioni.



Questo simbolo indica un possibile pericolo. È utilizzato per segnalare all'utente potenziali rischi di lesioni personali. Rispettare i messaggi di sicurezza evidenziati da questo simbolo per evitare da lesioni o rischi all'incolumità personale.

PERICOLO

PERICOLO indica una condizione immediata di pericolo, la quale, se non evitata, **può causare** seri rischi all'incolumità personale o gravi lesioni.

ATTENZIONE

ATTENZIONE indica una situazione di potenziale rischio che, se non evitata, **può provocare** morte o gravi infortuni.

 AVVERTENZA

AVVERTENZA indica una situazione di potenziale rischio, che, se non evitata, può provocare infortuni di lieve entità.

AVVERTENZA

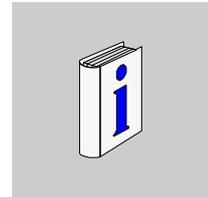
AVVERTENZA , senza il simbolo di allarme di sicurezza, indica una situazione di potenziale rischio che, se non evitata, può provocare danni alle apparecchiature.

NOTA

Manutenzione, riparazione, installazione e uso delle apparecchiature elettriche si devono affidare solo a personale qualificato. Schneider Electric non si assume alcuna responsabilità per qualsiasi conseguenza derivante dall'uso di questi prodotti.

Il personale qualificato possiede capacità e conoscenze relative alla struttura, al funzionamento e all'installazione di apparecchiature elettriche e ha ricevuto una formazione in materia di sicurezza che gli consente di riconoscere ed evitare i rischi del caso.

Informazioni su...



In breve

Scopo del documento

Questo documento permette all'utente di familiarizzare con le funzioni e le variabili di sistema disponibili nell' Modicon M258 Logic Controller. La libreria PLCSystem M258 contiene le funzioni e le variabili necessarie per ottenere informazioni e inviare comandi al sistema del controller.

Questo documento descrive le funzioni e le variabili dei tipi di dati della libreria PLCSystem M258.

Sono necessarie le seguenti conoscenze di base:

- Informazioni di base sulla funzionalità, la struttura e la configurazione dell'M258
- Nozioni di programmazione nei linguaggi FBD, LD, ST, IL o CFC
- Conoscenza delle variabili di sistema (variabili globali)

Nota di validità

Questo documento è stato aggiornato con la versione di SoMachine V3.0.

Documenti correlati

Titolo della documentazione	Numero di riferimento
Guida alla programmazione di Modicon M258 Logic Controller	EIO0000000402 (Eng), EIO0000000403 (Fre), EIO0000000404 (Ger), EIO0000000405 (Spa), EIO0000000406 (Ita), EIO0000000407 (Chs)

E' possibile scaricare queste pubblicazioni e tutte le altre informazioni tecniche dal sito www.schneider-electric.com.

Informazioni relative al prodotto

AVVERTENZA

PERDITA DI CONTROLLO

- Il progettista di qualsiasi schema di controllo deve prendere in considerazione le modalità di errore potenziali dei vari percorsi di controllo e, per alcune funzioni di controllo particolarmente critiche, deve fornire i mezzi per raggiungere uno stato di sicurezza durante e dopo un errore di percorso. Esempi di funzioni di controllo critiche sono ad esempio l'arresto di emergenza e gli stop di fine corsa, l'interruzione dell'alimentazione e il riavvio.
- Per le funzioni di controllo critiche occorre prevedere linee separate o ridondanti.
- Le linee di controllo di sistema possono comprendere collegamenti di comunicazione. È necessario fare alcune considerazioni sulle implicazioni di ritardi improvvisi nelle comunicazioni del collegamento.
- Osservare tutte le norme per la prevenzione degli incidenti e le normative di sicurezza locali.¹
- Prima della messa in servizio dell'apparecchiatura, controllare singolarmente e integralmente il funzionamento di ciascun controller.

Il mancato rispetto di queste istruzioni può provocare morte, gravi infortuni o danni alle apparecchiature.

¹ Per ulteriori informazioni, fare riferimento a NEMA ICS 1.1 (ultima edizione), "Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control" e a NEMA ICS 7.1 (ultima edizione), "Safety Standards for Construction and Guide for Selection, Installation, and Operation of Adjustable-Speed Drive Systems" o alla pubblicazione equivalente valida nel proprio paese.

AVVERTENZA

FUNZIONAMENTO ANOMALO DELL'APPARECCHIATURA

- Con questa apparecchiatura utilizzare esclusivamente il software approvato da Schneider Electric.
- Aggiornare il programma applicativo ogni volta che si cambia la configurazione dell'hardware fisico.

Il mancato rispetto di queste istruzioni può provocare morte, gravi infortuni o danni alle apparecchiature.

Commenti utente

Inviare eventuali commenti all'indirizzo e-mail techcomm@schneider-electric.com.

Variabili di sistema del M258

1

Panoramica

Questo capitolo:

- fornisce un'introduzione alle variabili di sistema (*vedi pagina 12*)
- descrive le variabili di sistema (*vedi pagina 18*) incluse nella libreria M258 PLCSystem

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
1.1	Variabili di sistema: definizione e uso	12
1.2	Strutture PLC_R e PLC_W	17
1.3	Strutture SERIAL_R e SERIAL_W	23
1.4	Strutture ETH_R e ETH_W	26
1.5	Struttura TM5_MODULE_R	32

1.1 Variabili di sistema: definizione e uso

Panoramica

Questa sezione definisce le variabili di sistema e ne descrive l'implementazione nell' Modicon M258 Logic Controller.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Variabili di sistema	13
Uso delle variabili di sistema	15

Variabili di sistema

Introduzione

Questa sezione descrive il modo in cui le variabili di sistema sono implementate per il controller. Queste variabili hanno i seguenti attributi:

- Le variabili di sistema permettono di accedere alle informazioni generali di sistema, di eseguire la diagnostica del sistema e di comandare azioni semplici.
- Le variabili di sistema sono variabili strutturate conformi alle definizioni e alle convenzioni per l'assegnazione dei nomi stabilite dalla norma IEC 61131. Per accedere a queste variabili si utilizza il nome simbolico IEC PLC_GVL.
- Alcune variabili PLC_GVL sono di sola lettura (ad esempio PLC_R), mentre altre sono di lettura/scrittura (ad esempio PLC_W).
- Le variabili di sistema sono dichiarate automaticamente come variabili globali. Queste variabili sono utilizzate a livello dell'intero sistema e devono essere gestite con attenzione poiché sono accessibili da qualsiasi POU (Program Organization Unit) in qualsiasi task.

Convenzione per l'assegnazione dei nomi delle variabili di sistema

Le variabili di sistema sono identificate da:

- un nome struttura, che rappresenta la categoria della variabile di sistema (ad es. PLC_R rappresenta una struttura di variabili di sola lettura utilizzate per la diagnostica dei controller).
- un set di nomi di componente, che identifica lo scopo della variabile (ad es. i_wVendorID rappresenta l'ID del fornitore del controller).

Si può accedere alle variabili digitando il nome della struttura delle variabili seguito dal nome del componente.

Le seguenti istruzioni sono un esempio di implementazione delle variabili di sistema:

```
VAR myCtr_Serial : DWORD; myCtr_ID : DWORD; myCtr_FramesRx :
UDINT; END_VAR

myCtr_Serial := PLC_R.i_dwSerialNumber; myCtr_ID :=
PLC_R.i_wVendorID; myCtr_FramesRx :=
SERIAL_R[0].i_udiFramesReceivedOK;
```

NOTA: Il nome qualificato completo della variabile di sistema utilizzata nell'esempio è PLC_GVL.PLC_R.i_wVendorID. Il nome PLC_GVL è implicito quando si dichiara una variabile utilizzando l'**Assistente di immissione**, tuttavia è possibile anche specificare il nome completo. La buona prassi di programmazione spesso richiede di specificare il nome qualificato completo della variabile nelle dichiarazioni.

Allocazione delle variabili di sistema

Durante la programmazione del controller vengono definiti due tipi di variabili di sistema:

- variabili identificate
- variabili non identificate

Le variabili identificate:

- risiedono in una posizione fissa di un'area di memoria %MW statica:
 - da %MW60000 a %MW60199 per le variabili di sistema in sola lettura
 - da %MW62000 a %MW62199 per le variabili di sistema in lettura/scrittura
- sono accessibili tramite richieste Modbus TCP, Modbus seriale e Ethernet IP sia in stato RUNNING che in stato STOPPED
- sono utilizzate nei programmi SoMachine in base alla convenzione `structure_name.component_name` spiegata in precedenza (gli indirizzi %MW compresi tra 0 e 59999 consentono un accesso diretto; gli indirizzi maggiori sono considerati fuori intervallo da SoMachine e consentono l'accesso solo tramite la convenzione `structure_name.component_name`).

Le variabili non identificate:

- non sono fisicamente collocate nell'area %MW
- non sono accessibili da bus di campo o richieste di rete a meno che non vengano collocate nella tabella di rilocazione; solo a quel punto è possibile accedervi sia nello stato RUNNING che nello stato STOPPED. La tabella di rilocazione usa le seguenti aree %MW dinamiche:
 - da %MW60200 a %MW61999 per le variabili di sistema in sola lettura
 - da %MW62200 a %MW63999 per le variabili di lettura/scrittura
- sono utilizzate nei programmi SoMachine in base alla convenzione `structure_name.component_name` spiegata in precedenza (gli indirizzi %MW compresi tra 0 e 59999 consentono un accesso diretto; gli indirizzi maggiori sono considerati fuori intervallo da SoMachine e consentono l'accesso solo tramite la convenzione `structure_name.component_name`).

Uso delle variabili di sistema

Introduzione

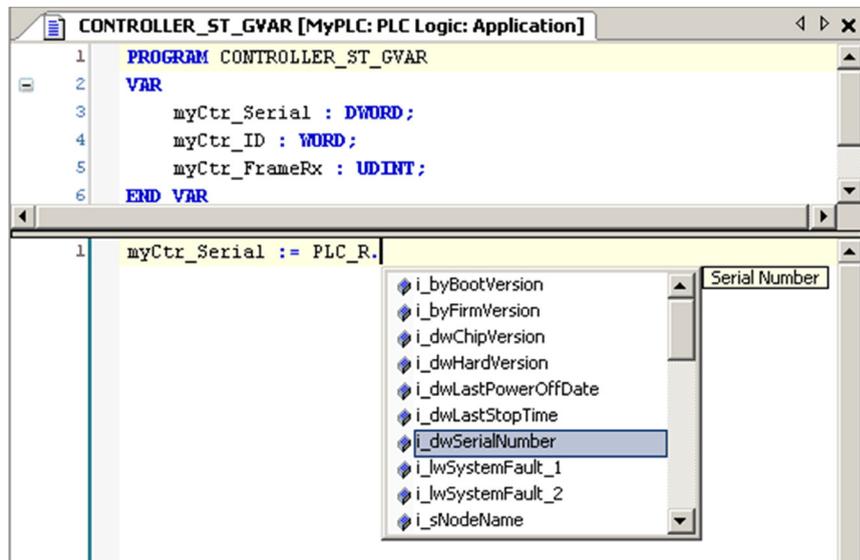
Questa sezione descrive la procedura necessaria per programmare e utilizzare le variabili di sistema in SoMachine.

Le variabili di sistema possono essere utilizzate in tutte le POU (Program Organization Units) dell'applicazione.

Non è necessario dichiarare le variabili di sistema nella GVL. Esse sono dichiarate automaticamente dalla libreria di sistema del controller.

Uso delle variabili di sistema in una POU

Nella **POU**, immettere il nome della struttura della variabile di sistema (PLC_R, PLC_W...) seguito da un punto. Le **variabili di sistema** sono visualizzate nell'**Accesso facilitato**. Si può selezionare la variabile desiderata oppure immettere il nome completo manualmente.



NOTA: SoMachine dispone di una funzionalità di completamento automatico. Nell'esempio precedente, una volta digitato il nome della struttura PLC_R., SoMachine propone una serie di possibili nomi/variabili del componente in un menu a comparsa.

Esempio

Il seguente esempio illustra l'uso di alcune variabili di sistema:

```
VAR myCtr_Serial : DWORD; myCtr_ID : WORD; myCtr_FramesRx :  
UDINT; END_VAR  
  
myCtr_Serial := PLC_R.i_dwSerialNumber; myCtr_ID :=  
PLC_R.i_wVendorID; myCtr_FramesRx :=  
SERIAL_R[0].i_udiFramesReceivedOK;
```

1.2 Strutture PLC_R e PLC_W

Panoramica

In questa sezione sono elencate e descritte le **variabili di sistema** incluse nelle strutture PLC_R e PLC_W.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
PLC_R: variabili di sistema in sola lettura del controller	18
PLC_W: variabili di sistema in lettura/scrittura del controller	22

PLC_R: variabili di sistema in sola lettura del controller

Struttura delle variabili

La seguente tabella descrive i parametri della variabile di sistema PLC_R (tipo PLC_R_STRUCT):

%MW	Nome var	Tipo	Commento
60000	i_wVendorID	WORD	ID fornitore del controller. 101A hex= Schneider Electric
60001	i_wProductID	WORD	ID codice prodotto del controller. NOTA: L'ID fornitore e l'ID prodotto sono i componenti dell'ID target del controller visualizzato nella vista Impostazioni di comunicazione (ID target = 101A XXXX hex).
60002	i_dwSerialNumber	DWORD	Numero di serie del controller
60004	i_byFirmVersion[0..3]	ARRAY[0..3] OF BYTE	Versione firmware del controller [aa.bb.cc.dd]: <ul style="list-style-type: none"> ● i_byFirmVersion[0]= aa ● ... ● i_byFirmVersion[3]= dd
60006	i_byBootVersion[0..3]	ARRAY[0..3] OF BYTE	Versione di avvio del controller [aa.bb.cc.dd]: <ul style="list-style-type: none"> ● i_byBootVersion[0]= aa ● ... ● i_byBootVersion[3]= dd
60008	i_dwHardVersion	DWORD	Versione dell'hardware del controller.
60010	i_dwChipVersion	DWORD	Versione del coprocessore del controller.
60012	i_wStatus	PLC_R_STATUS (vedi pagina 52)	Stato del controller.
60013	i_wBootProjectStatus	PLC_R_BOOT_PROJECT_STATUS (vedi pagina 50)	Restituisce le informazioni sull'applicazione di avvio memorizzata nella memoria FLASH.
60014	i_wLastStopCause	PLC_R_STOP_CAUSE (vedi pagina 53)	Causa dell'ultima transizione da RUN a un altro stato.
60015	i_wLastApplicationError	PLC_R_APPLICATION_ERROR (vedi pagina 49)	Causa dell'ultima eccezione del controller.

%MW	Nome var	Tipo	Commento
60016	i_lwSystemFault_1	LWORD	<p>Il campo bit FFFF FFFF FFFF FFFF hex indica che non sono stati rilevati errori. Un bit al livello basso significa che è stato rilevato un errore:</p> <ul style="list-style-type: none"> ● bit 0 = errore rilevato di Expert integrato. Vedere i_wIOStatus1 per la diagnostica ● bit 1 = errore I/O TM5 rilevato. Vedere i_wIOStatus2 per la diagnostica ● bit 2 = errore Ethernet 0 rilevato ● bit 3 = errore Serial 0 rilevato ● bit 4 = errore CAN 0 rilevato ● bit 5 = errore CAN 1 rilevato ● bit 6 = errore modulo bus di interfaccia 0 rilevato ● bit 7 = errore modulo bus di interfaccia 1 rilevato
60020	i_lwSystemFault_2	LWORD	Non utilizzati.
60024	i_wIOStatus1	PLC_R_IO_STATUS (vedi pagina 51)	Stato I/O Esperto integrato.
60025	i_wIOStatus2	PLC_R_IO_STATUS (vedi pagina 51)	Stato I/O TM5.
60026	i_wClockBatteryStatus	WORD	<p>Stato della carica della batteria dell'orologio in tempo reale:</p> <ul style="list-style-type: none"> ● 0000 hex = batteria scarica ● FFFF hex = carica della batteria ok
60028	i_dwAppliSignature1	DWORD	<p>Prima DWORD di firma a 4 DWORD (16 byte in totale). La firma dell'applicazione è generata dal software durante la compilazione.</p>
60030	i_dwAppliSignature2	DWORD	<p>Seconda DWORD di firma a 4 DWORD (16 byte in totale). La firma dell'applicazione è generata dal software durante la compilazione.</p>
60032	i_dwAppliSignature3	DWORD	<p>Terza DWORD di firma a 4 DWORD (16 byte in totale). La firma dell'applicazione è generata dal software durante la compilazione.</p>
60034	i_dwAppliSignature4	DWORD	<p>Quarta DWORD di firma a 4 DWORD (16 byte in totale). La firma dell'applicazione è generata dal software durante la compilazione.</p>

n.d.	i_sVendorName	STRING (31)	Nome del fornitore: "Schneider Electric".
n.d.	i_sProductRef	STRING (31)	Codice prodotto del controller.
n.d.	i_sNodeName	STRING (31)	Nome del nodo sulla rete SoMachine
n.d.	i_dwiLastStopTime	DWORD	L'ora dell'ultimo STOP rilevato in secondi con inizio il 1° gennaio 1970 alle 00:00.
n.d.	i_dwLastPowerOffDate	DWORD	La data e l'ora dell'ultimo spegnimento rilevato in secondi con inizio il 1° gennaio 1970 alle 00:00. NOTA: È possibile convertire questo valore in data e ora usando la funzione <code>SysTimeRtcConvertUtcToDate</code> . Per maggiori informazioni sulla conversione in ora e data, fare riferimento alla Guida della libreria System (<i>vedi SoMachine, Accesso e configurazione orologio in tempo reale (RTC), Guida della libreria SysTime</i>).
n.d.	i_uiEventsCounter	UINT	Numero di eventi esterni rilevati sugli ingressi configurati per il rilevamento degli eventi esterni dall'ultimo avvio a freddo. Azzeramento con un riavvio a freddo o con il comando <code>PLC_W.q_wResetCounterEvent</code> .
n.d.	i_wTerminalPortStatus	PLC_R_TERMINAL_PORT_STATUS (vedi pagina 54)	Stato della porta di programmazione USB (USB di tipo mini B).
n.d.	i_wUSBHostStatus	PLC_R_USB_HOST_STATUS (vedi pagina 55)	Stato della porta host USB (USB A).
n.d.	i_wUsrFreeFileHdl	WORD	Numero di handle di file che possono essere aperti. Un handle di file contiene le risorse allocate dal sistema quando si apre un file.
n.d.	i_udiUsrFsTotalBytes	UDINT	Dimensioni totali della memoria del FileSystem utente (in byte). Dimensioni della memoria flash per la directory <code>"/usr/</code> .
n.d.	i_udiUsrFsFreeBytes	UDINT	Dimensioni della memoria libera del FileSystem utente (in byte).

n.d.	i_uiTM5BusState	UINT	<p>Campo bit stato del bus TM5:</p> <ul style="list-style-type: none"> ● bit 0..3 = non utilizzati ● bit 4= driver bus TM5 disponibile ● bit 5 = hardware bus TM5 trovato ● bit 6 = configurazione bus TM5 eseguita correttamente ● bit 7 = bus TM5 operativo ● bit 8 = non utilizzati ● bit 9 = errore rilevato durante la configurazione del bus TM5 ● bit 10..15 = non utilizzati
n.d.	i_uiTM5SyncErrCnt	UINT	<p>Numero di frame sincroni non validi rilevati sul bus TM5. Reset con il comando PLC_W.q_wResetTM5Counters e alla disinserzione.</p>
n.d.	i_uiTM5AsynErrCnt	UINT	<p>Numero di frame asincroni non validi rilevati sul bus TM5. Reset con il comando PLC_W.q_wResetTM5Counters e alla disinserzione.</p>
n.d.	i_uiTM5BreakCnt	UINT	<p>Numero di reset del bus TM5 rilevati. Reset con il comando PLC_W.q_wResetTM5Counters e alla disinserzione.</p>
n.d.	i_uiTM5TopoChangedCnt	UINT	<p>Numero di modifiche nella topologia del bus TM5. Reset con il comando PLC_W.q_wResetTM5Counters e alla disinserzione.</p>
n.d.	i_uiTM5BusCycleCnt	UINT	<p>Numero di cicli del bus TM5 dall'avvio a freddo. Reset con il comando PLC_W.q_wResetTM5Counters e alla disinserzione.</p>
n.d.	i_wTM5BrokendownSlot	WORD	<p>00..FE hex= numero slot di un modulo TM5 non funzionante. FF hex = tutti i moduli TM5 segnalano che sono funzionanti.</p>

NOTA: n/d significa che non vi è una mappatura %MW predefinita per questa variabile di sistema.

PLC_W: variabili di sistema in lettura/scrittura del controller

Struttura delle variabili

La seguente tabella descrive i parametri della variabile di sistema PLC_W (tipo PLC_W_STRUCT):

%MW	Nome var	Tipo	Commento
n.d.	q_wResetCounterEvent	WORD	La transizione da 0 a 1 azzerà il contatore eventi (PLC_R.i_uiEventsCounter). Per azzerare nuovamente il contatore occorre scrivere questo registro a 0, in modo che possa verificarsi un'altra transizione da 0 a 1.
n.d.	q_wResetTM5counters	WORD	La transizione da 0 a 1 azzerà tutti i contatori TM5 delle variabili di sistema strutturate PLC_R (da PLC_R.i_uiTM5SyncErrCnt a PLC_R.i_uiTM5BusCycleCnt). Per azzerare di nuovo i contatori, è necessario scrivere questo registro a 0 prima che possa verificarsi un'altra transizione da 0 a 1.
n.d.	q_uiOpenPLCControl	UINT	Quando il valore passa da 0 a 6699, viene eseguito il comando precedentemente scritto nel PLC_W.q_wPLCControl successivo.
n.d.	q_wPLCControl	PLC_W_COMMAND (vedi pagina 56)	Comando di RUN / STOP del controller eseguito quando il valore della variabile di sistema PLC_R.q_uiOpenPLCControl passa da 0 a 6699.

NOTA: n/d significa che non vi è una mappatura %MW predefinita per questa variabile di sistema.

1.3 Strutture SERIAL_R e SERIAL_W

Panoramica

Questa sezione elenca e descrive le diverse **Variabili di sistema** incluse nelle strutture SERIAL_R e SERIAL_W.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
SERIAL_R[0..1]: variabili di sistema in sola lettura della linea seriale	24
SERIAL_W[0..1]: variabili di sistema in lettura/scrittura della linea seriale	25

SERIAL_R[0..1]: variabili di sistema in sola lettura della linea seriale

Introduzione

SERIAL_R è un array di due tipi SERIAL_R_STRUCT. Ogni elemento dell'array restituisce le **variabili di sistema** diagnostiche per la linea seriale corrispondente.

Per M258:

- Serial_R[0] si riferisce alla linea seriale integrata
- Serial_R[1] si riferisce al modulo PCI seriale opzionale, se installato

Struttura della variabile

La tabella seguente descrive i parametri della variabile di sistema

SERIAL_R[0..1]:

%MW	Nome della variabile	Tipo	Commento
Linea seriale			
n.d.	i_udiFramesTransmittedOK	UDINT	Numero di frame trasmessi correttamente.
n.d.	i_udiFramesReceivedOK	UDINT	Numero di frame ricevuti senza nessun errore rilevato.
n.d.	i_udiRX_MessagesError	UDINT	Numero di frame ricevuti con errori rilevati (checksum, parità).
Specifici di Modbus			
n.d.	i_uiSlaveExceptionCount	UINT	Numero di risposte alle eccezioni Modbus restituite dal controller.
n.d.	i_udiSlaveMsgCount	UINT	Numero di messaggi ricevuti dal master e indirizzati al controller.
n.d.	i_uiSlaveNoRespCount	UINT	Numero di richieste broadcast Modbus ricevute dal controller.
n.d.	i_uiSlaveNakCount	UINT	Non usato
n.d.	i_uiSlaveBusyCount	UINT	Non usato
n.d.	i_uiCharOverrunCount	UINT	Overrun numero di caratteri.

NOTA: n/d significa che non vi è una mappatura %MW predefinita per questa variabile di sistema.

NOTA:

I contatori SERIAL_R vengono azzerati nei seguenti casi:

- download
- reset del controller
- comando SERIAL_W[x].q_wResetCounter
- comando di azzeramento tramite codice funzione di richiesta Modbus 8.

SERIAL_W[0..1]: variabili di sistema in lettura/scrittura della linea seriale

Introduzione

SERIAL_W è un array di due tipi SERIAL_W_STRUCT. Ogni elemento dell'array forza le variabili di sistema **SERIAL_R** in modo che la corrispondente linea seriale venga resettata.

Per M258:

- Serial_W[0] fa riferimento alla linea seriale integrata
- Serial_W[1] fa riferimento alla linea seriale PCI

Struttura delle variabili

La tabella seguente descrive i parametri della variabile di sistema

SERIAL_W[0..1]:

%MW	Nome var	Tipo	Commento
n.d.	q_wResetCounter	WORD	La transizione da 0 a 1 azzerata tutti i contatori SERIAL_R[0..1]. Per azzerare di nuovo i contatori, è necessario scrivere questo registro a 0 prima che possa verificarsi un'altra transizione da 0 a 1.

NOTA: n/d significa che non vi è una mappatura %MW predefinita per questa variabile di sistema.

1.4 Strutture ETH_R e ETH_W

Panoramica

In questa sezione sono elencate e descritte le **variabili di sistema** incluse nelle strutture ETH_R e ETH_W.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
ETH_R: variabili di sistema in sola lettura porta Ethernet	27
ETH_W: variabili di sistema lettura/scrittura porta Ethernet	31

ETH_R: variabili di sistema in sola lettura porta Ethernet

Struttura delle variabili

La seguente tabella descrive i parametri della variabile di sistema ETH_R (tipo ETH_R_STRUCT):

%MW	Nome var	Tipo	Commento
60050	i_byIPAddress[0..3]	ARRAY[0.00,3] OF BYTE	Indirizzo IP [aaa.bbb.ccc.ddd]: ● i_byIPAddress[0]= aaa ● ... ● i_byIPAddress[3]= ddd
60052	i_bySubNetMask[0..3]	ARRAY[0.00,3] OF BYTE	Subnet Mask [aaa.bbb.ccc.ddd]: ● i_bySub-netMask[0]= aaa ● ... ● i_bySub-netMask[3]= ddd
60054	i_byGateway[0..3]	ARRAY[0.00,3] OF BYTE	Indirizzo gateway [aaa.bbb.ccc.ddd]: ● i_byGateway[0]= aaa ● ... ● i_byGateway[3]= ddd
60056	i_byMACAddress[0..5]	ARRAY[0.00,5] OF BYTE	Indirizzo MAC [aa.bb.cc.dd.ee.ff]: ● i_byMACAddress[0]= aa ● ... ● i_byMACAddress[5]= ff
60059	i_sDeviceName	STRING(16)	Nome utilizzato per ottenere l'indirizzo IP dal server.
n.d.	i_wIpMode	ETH_R_IP_MODE (vedi pagina 58)	Metodo usato per ottenere un indirizzo IP
n.d.	i_byFDRServerIPAddress[0..3]	ARRAY[0..3] OF BYTE	L'indirizzo IP [aaa.bbb.ccc.ddd] del server DHCP o BootP: ● i_byFDRServerIPAddress[0]= aaa ● ... ● i_byFDRServerIPAddress[3]= ddd Pari a 0.0.0.0 se viene utilizzato l'IP memorizzato o predefinito.
n.d.	i_udiOpenTcpConnections	UDINT	Numero di connessioni TCP aperte.

%MW	Nome var	Tipo	Commento
n.d.	i_wFrameSendingProtocol	ETH_R_FRAME_PROTOCOL (vedi pagina 59)	Protocollo Ethernet configurato per l'invio di frame (IEEE 802.3 o Ethernet II).
n.d.	i_udiFramesTransmittedOK	UDINT	Numero di frame trasmessi correttamente. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiFramedReceivedOK	UDINT	Numero di frame ricevuti correttamente. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiTransmitBufferErrors	UDINT	Numero di frame trasmessi con errori rilevati. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiReceiveBufferErrors	UDINT	Numero dei frame ricevuti con errori rilevati. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_wPortALinkStatus	ETH_R_PORT_LINK_STATUS (vedi pagina 61)	Direzione del collegamento della porta Ethernet (0= indietro o 1= avanti).
n.d.	i_wPortASpeed	ETH_R_PORT_SPEED (vedi pagina 62)	Velocità di rete della porta Ethernet (10Mb/s o 100Mb/s).
n.d.	i_wPortADuplexStatus	ETH_R_PORT_DUPLEX_STATUS (vedi pagina 60)	Stato duplex della porta Ethernet (0= Half duplex o 1= Full duplex).
n.d.	i_udiPortACollisions	UDINT	Numero di frame coinvolti in una o più collisioni e successivamente trasmessi correttamente. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_wPortAIPStatus	ETH_R_PORT_IP_STATUS (vedi pagina 63)	Stato stack porta Ethernet TCP/IP.
Specifiche Modbus TCP/IP			

%MW	Nome var	Tipo	Commento
n.d.	i_udiModbusMessageTransmitted	UDINT	Numero di messaggi Modbus trasmessi. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiModbusMessageReceived	UDINT	Numero di messaggi Modbus ricevuti. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiModbusErrorMessage	UDINT	Messaggi di errore rilevati Modbus trasmessi e ricevuti. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_byMasterIpTimeouts	BYTE	Contatore eventi di timeout Maser Modbus TCP Ethernet. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_byMasterIpLost	BYTE	Stato collegamento master Modbus TCP Ethernet: 0 = collegamento OK, 1 = collegamento perso.

Specifiche EtherNet/IP

n.d.	i_udiETHIP_IOMessagingTransmitted	UDINT	Frame EtherNet/IP Classe 1 trasmessi. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiETHIP_IOMessagingReceived	UDINT	Frame EtherNet/IP Classe 1 ricevuti. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiUCMM_Request	UDINT	Messaggi non connessi EtherNet/IP ricevuti. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.

n.d.	i_udiUCMM_Error	UDINT	Messaggi non connessi non validi EtherNet/IP ricevuti. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiClass3_Request	UDINT	Richieste EtherNet/IP Classe 3 ricevute. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_udiClass3_Error	UDINT	Richieste EtherNet/IP Classe 3 non valide ricevute. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_uiAssemblyInstanceInput	UINT	Numero istanza gruppo ingressi. Per maggiori informazioni consultare la guida alla programmazione appropriata del controller.
n.d.	i_uiAssemblyInstanceInputSize	UINT	Dimensioni istanza gruppo ingressi. Per maggiori informazioni consultare la guida alla programmazione appropriata del controller.
n.d.	i_uiAssemblyInstanceOutput	UINT	Numero istanza gruppo uscite. Per maggiori informazioni consultare la guida alla programmazione appropriata del controller.
n.d.	i_uiAssemblyInstanceOutputSize	UINT	Dimensioni istanza gruppo uscite. Per maggiori informazioni consultare la guida alla programmazione appropriata del controller.
n.d.	i_uiETHIP_ConnectionTimeouts	UINT	Numero di timeout di connessione. Azzeramento all'accensione o con il comando di reset ETH_W.q_wResetCounter.
n.d.	i_ucEipRunIdle	ETH_R_RUN_IDLE (vedi pagina 64)	Contrassegno Run (value=1)/Idle(value=0) per connessione 1 di classe EtherNet/IP 1.

NOTA: n/d significa che non vi è una mappatura %MW predefinita per questa variabile di sistema.

ETH_W: variabili di sistema lettura/scrittura porta Ethernet

Struttura delle variabili

La seguente tabella descrive i parametri della variabile di sistema ETH_W (tipo ETH_W_STRUCT):

%MW	Nome var	Tipo	Commento
n.d.	q_wResetCounter	WORD	La transizione da 0 a 1 azzerata tutti i contatori ETH_R. Per azzerare nuovamente i contatori occorre scrivere questo registro a 0, in modo che possa verificarsi un'altra transizione da 0 a 1.

NOTA: n/d significa che non vi è una mappatura %MW predefinita per questa variabile di sistema.

1.5 Struttura TM5_MODULE_R

TM5_MODULE_R[1..254]: Variabili di sistema di sola lettura dei moduli TM5

Introduzione

TM5_MODULE_R è un array di 254 moduli di tipo TM5_MODULE_R_STRUCT. Ogni elemento dell'array restituisce la diagnostica delle **variabili di sistema** per il modulo TM5 corrispondente.

For M258:

- TM5_MODULE_R[1] si riferisce al modulo TM5 1
- ...
- TM5_MODULE_R[254] si riferisce al modulo TM5 254

Struttura della variabile

La tabella seguente descrive i parametri della variabile di sistema TM5_MODULE_R[1..254]:

%MW	Nome della variabile	Tipo	Commento
n.d.	i_wVendorID	WORD	ID fornitore modulo TM5 della destinazione.
n.d.	i_wProductID	WORD	ID tipo di modulo TM5 della destinazione.
n.d.	i_dwSerialNumber	DWORD	numero di serie del modulo TM5.
n.d.	i_wFirmVersion	WORD	versione firmware del modulo TM5.
n.d.	i_wBootVersion	WORD	versione di avvio del modulo TM5.
n.d.	i_wModuleState	TM5_MODULE_STATE (vedi pagina 65)	Descrive lo stato del modulo TM5. Il modulo è operativo quando viene restituito TM5_ACTIVE.

NOTA: n/d significa che non vi è una mappatura %MW predefinita per questa variabile di sistema.

Funzioni di sistema del M258

2

Panoramica

Questo capitolo descrive le funzioni incluse nella libreria M258 PLCSystem.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
2.1	Funzioni di lettura M258	34
2.2	Funzioni di scrittura del M258	42

2.1 Funzioni di lettura M258

Panoramica

Questa sezione descrive le funzioni di lettura incluse nella libreria M258 PLCSystem.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
DM72FGetImmediateInput: lettura dell'ingresso di un I/O Expert integrato	35
getTM5Delay: Numero di cicli del bus TM5 senza scambio valido	36
IsFirstMastColdCycle: indica se il ciclo è il primo ciclo MAST dopo un avvio a freddo	38
IsFirstMastCycle: indica se il ciclo è il primo ciclo MAST	39
IsFirstMastWarmCycle: indica se il ciclo è il primo ciclo MAST dopo un avvio a caldo	41

DM72FGetImmediateInput: lettura dell'ingresso di un I/O Expert integrato

Descrizione della funzione

Questa funzione è applicabile al **blocco di I/O Expert integrato** DM72F0 e DM72F1. Restituisce il valore fisico corrente dell'ingresso, che può essere diverso dal valore logico corrente di quell'ingresso. Il valore della variabile per quell'ingresso non cambia fino al ciclo di bus successivo.

Rappresentazione grafica



Rappresentazione IL e ST

Per la rappresentazione generale in linguaggio IL o ST, fare riferimento al capitolo *Rappresentazione di funzioni e blocchi funzione (vedi pagina 77)*.

Descrizione della variabile di I/O

La tabella seguente descrive le variabili di ingresso:

Ingresso	Tipo	Commento
Blocco	INT	Blocco di destinazione: <ul style="list-style-type: none"> ● 0= DM72F0 ● 1= DM72F1
Ingresso	INT	Ingresso di destinazione del blocco. 0..6= DI0..DI6

Nella seguente tabella sono descritte le variabili di uscita

Uscita	Tipo	Commento
DM72FGetImmediateInput	BOOL	Valore dell'ingresso <Input> del blocco <block>= FALSE/TRUE.
Error	BOOL	FALSE= operazione ok. TRUE= rilevato errore operazione, la funzione restituisce un valore non valido.
ErrID	IMMEDIATE_FUNC_ERR_TYPE	Codice errore operazione rilevato quando Error è TRUE.

getTM5Delay: Numero di cicli del bus TM5 senza scambio valido

Descrizione della funzione

Questa funzione restituisce il numero di cicli del bus TM5 senza nessuno scambio valido con un modulo di destinazione TM5.

NOTA: Per la diagnostica del modulo TM5, vedere la variabile di sistema TM5_MODULE_R (vedi pagina 32).

Rappresentazione grafica



Rappresentazione IL e ST

Per la rappresentazione generale in linguaggio IL o ST, fare riferimento al capitolo *Rappresentazione di funzioni e blocchi funzione* (vedi pagina 77).

Descrizione della variabile di I/O

La tabella seguente descrive le variabili di ingresso:

Ingresso	Tipo	Commento
ID nodo	DINT	<p>NodeNumber del modulo TM5 di destinazione (per ottenere il numero del nodo, verificare il valore nell'editor del modulo TM5 nella schermata Configurazione I/O).</p>

Nella seguente tabella sono descritte le variabili di uscita

Uscita	Tipo	Commento
getTM5Delay	USINT	<p>Questa variabile può assumere i valori seguenti:</p> <ul style="list-style-type: none"> ● 0= OK ● [1..3]= da 1 a 3 cicli senza scambio valido] ● -1= più di 3 cicli senza scambio valido o parametro non valido

Esempio

Il seguente esempio spiega come ottenere il ritardo del primo modulo TM5:

```
VAR
    delay : DINT;
    //Slot ID is 1 for the first TM5 module
    slot_ID : USINT := 1;
END_VAR
delay = getTM5Delay(slot_ID);
```

IsFirstMastColdCycle: indica se il ciclo è il primo ciclo MAST dopo un avvio a freddo

Descrizione della funzione

Questa funzione restituisce TRUE durante il primo ciclo MAST dopo un avvio a freddo (primo ciclo dopo un download o dopo un riavvio a freddo).

Rappresentazione grafica



Rappresentazione IL e ST

Per la rappresentazione generale in linguaggio IL o ST, fare riferimento al capitolo *Rappresentazione di funzioni e blocchi funzione (vedi pagina 77)*.

Descrizione delle variabili I/O

Nella tabella seguente è descritta la variabile di uscita:

Uscita	Tipo	Commento
IsFirstMastColdCycle	BOOL	TRUE durante il primo ciclo del task MAST dopo un avvio a freddo.

Esempio

Fare riferimento alla funzione `IsFirstMastCycle` (vedi pagina 39).

IsFirstMastCycle: indica se il ciclo è il primo ciclo MAST

Descrizione della funzione

Questa funzione restituisce TRUE durante il primo ciclo MAST dopo un avvio.

Rappresentazione grafica



Rappresentazione IL e ST

Per la rappresentazione generale in linguaggio IL o ST, fare riferimento al capitolo *Rappresentazione di funzioni e blocchi funzione (vedi pagina 77)*.

Descrizione della variabile di I/O

Uscita	Tipo	Commento
IsFirstMastCycle	BOOL	TRUE durante il primo ciclo del task MAST dopo un avvio.

Esempio

Questo esempio descrive le tre funzioni `IsFirstMastCycle`, `IsFirstMastColdCycle` e `IsFirstMastWarmCycle` utilizzate insieme.

Questo esempio deve essere usato nel task MAST, altrimenti potrebbe essere eseguito più volte o non essere mai eseguito (un task aggiuntivo potrebbe essere richiamato più volte o non essere richiamato durante un ciclo del task MAST):

```
VAR MyIsFirstMastCycle : BOOL; MyIsFirstMastWarmCycle : BOOL;
MyIsFirstMastColdCycle : BOOL; END_VAR

MyIsFirstMastWarmCycle := IsFirstMastWarmCycle();
MyIsFirstMastColdCycle := IsFirstMastColdCycle();
MyIsFirstMastCycle := IsFirstMastCycle();

IF (MyIsFirstMastWarmCycle) THEN
  (*Questo è il primo ciclo Mast dopo un avvio a caldo: tutte le
  variabili sono impostate ai valori di inizializzazione eccetto
  le variabili ritentive*)
  (*=> inizializzare le variabili necessarie in modo che
  l'applicazione venga eseguita come previsto in questo caso*)
END_IF;
```

```
IF (MyIsFirstMastColdCycle) THEN
  (*Questo è il primo ciclo Mast dopo un avvio a freddo: tutte
  le variabili sono impostate ai valori di inizializzazione
  incluse le variabili ritentive*)
  (*=> inizializzare le variabili necessarie in modo che
  l'applicazione venga eseguita come previsto in questo caso*)
END_IF;
IF (MyIsFirstMastCycle) THEN
  (*Questo è il primo ciclo Mast dopo un avvio, a caldo o a
  freddo, e dopo i comandi STOP/RUN*)
  (*=> inizializzare le variabili necessarie in modo che
  l'applicazione venga eseguita come previsto in questo caso*)
END_IF;
```

IsFirstMastWarmCycle: indica se il ciclo è il primo ciclo MAST dopo un avvio a caldo

Descrizione della funzione

Questa funzione restituisce TRUE durante il primo ciclo MAST dopo un avvio a caldo.

Rappresentazione grafica



Rappresentazione IL e ST

Per la rappresentazione generale in linguaggio IL o ST, fare riferimento al capitolo *Rappresentazione di funzioni e blocchi funzione (vedi pagina 77)*.

Descrizione delle variabili I/O

Nella tabella seguente è descritta la variabile di uscita:

Uscita	Tipo	Commento
IsFirstMastWarmCycle	BOOL	TRUE durante il primo ciclo del task MAST dopo un avvio a caldo.

Esempio

Fare riferimento alla funzione `IsFirstMastCycle` (vedi pagina 39).

2.2 Funzioni di scrittura del M258

Panoramica

Questa sezione descrive le funzioni di scrittura incluse nella libreria M258 PLCSystem.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

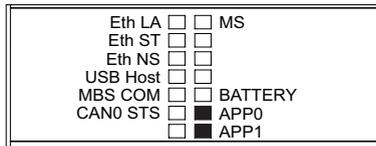
Argomento	Pagina
SetLEDBehaviour: determina il comportamento di un LED	43
SetRTCDrift: regolazione settimanale dell'orologio in tempo reale	45

SetLEDBehaviour: determina il comportamento di un LED

Descrizione della funzione

Questa funzione controlla i LED di diagnostica APP0 e APP1.

Nella figura riportata di seguito sono illustrati i LED sul display del pannello frontale:



Rappresentazione grafica



Rappresentazione IL e ST

Per la rappresentazione generale in linguaggio IL o ST, fare riferimento al capitolo *Rappresentazione di funzioni e blocchi funzione (vedi pagina 77)*.

Descrizione delle variabili I/O

Nella tabella seguente sono descritti i parametri d'ingresso:

Ingressi	Tipo	Commento
LedId	LED_ID (vedi pagina 67)	ID del LED applicazione.
LedColor	LED_COLOR (vedi pagina 70)	Colore del LED applicazione.
LedBhv	LED_BHV (vedi pagina 68)	Modalità del LED applicazione.

Nella tabella seguente è descritta la variabile di uscita:

Uscita	Tipo	Commento
SetLEDBehaviour	LED_BHV_ERROR (vedi pagina 69)	Restituisce NO_ERROR (00 hex) se il comando viene eseguito correttamente, altrimenti restituisce il codice ID dell'errore rilevato.

Esempio

Questo esempio mostra come comandare il LED APP0 perché si accenda di luce verde:

```

VAR
    myLEDStatus : LED_BHV_ERROR;
    myLED : LED_ID := LED_0;
    myLEDColor : LED_COLOR := LED_GREEN;
    myLEDMode : LED_BHV := LED_ON;
END_VAR
myLEDStatus := SetLedBehaviour(myLED, myLEDColor, myLEDMode);
    
```

NOTA: I colori dei LED sono controllati separatamente e possono essere combinati; spegnere il colore corrente prima di accendere il nuovo. Nella tabella seguente è illustrato un esempio di sequenza di comandi `SetLedBehaviour` con il relativo comportamento dei LED:

passo	LedId	LedColor	LedBhv	Modalità IAmpeggiamento VERDE	Modalità lampeggiamento ROSSO
1	LED_0	-	-	OFF	OFF
2	LED_0	LED_GREEN	LED_ON	ON	OFF
3	LED_0	LED_GREEN	LED_OFF	OFF	OFF
4	LED_0	LED_RED	LED_ON	OFF	ON

SetRTCDrift: regolazione settimanale dell'orologio in tempo reale

Descrizione della funzione

Questa funzione aggiunge o sottrae al tempo indicato dall'orologio in tempo reale una quantità di secondi specificata, una volta alla settimana nel giorno specificato, ora:minuto

NOTA: La funzione SetRTCDrift deve essere programmato per essere eseguito solo durante il primo ciclo Mast.

Rappresentazione grafica



Rappresentazione IL e ST

Per la rappresentazione generale in linguaggio IL o ST, fare riferimento al capitolo *Rappresentazione di funzioni e blocchi funzione (vedi pagina 77)*.

Descrizione delle variabili I/O

Nella tabella seguente sono descritti i parametri d'ingresso:

Ingressi	Tipo	Commento
RTCDrift	SINT(-29..29)	Correzione in secondi (-29 ... +29)
Giorno	DAY_OF_WEEK (vedi pagina 72)	Giorno della settimana in cui il giorno verrà eseguito.
Ora	HOUR (vedi pagina 73)	Ora in cui viene eseguita la modifica.
Minuto	MINUTE (vedi pagina 74)	Minuto in cui viene eseguita la modifica.

NOTA: Se i valori immessi per RTCDrift, Day, Hour, Minute superano i valori limite, il firmware del controller imposterà tutti i valori ai valori massimi.

Nella tabella seguente è descritta la variabile di uscita:

Uscita	Tipo	Commento
SetRTCDrift	RTCSETDRIFT_ERROR (vedi pagina 71)	Restituisce RTC_OK (00 hex) se il comando viene eseguito correttamente, altrimenti restituisce il codice ID dell'errore rilevato.

Esempio

In questo esempio, la funzione viene richiamata solo una volta durante il ciclo del primo task MAST, vengono aggiunti 20 secondi all'RTC ogni martedì alle 5:45 a.m.:

```
VAR MyRTCDrift : SINT (-29..29) := 0; MyDay : DAY_OF_WEEK;  
MyHour : HOUR; MyMinute : MINUTE; END_VAR  
  
IF IsFirstMastCycle() THEN MyRTCDrift := 20; MyDay := TUESDAY;  
MyHour := 5; MyMinute := 45; SetRTCDrift(MyRTCDrift, MyDay,  
MyHour, MyMinute); END_IF
```

Libreria M258 PLCSystem - Tipi di dati

3

Panoramica

Questo capitolo descrive i **tipi di dati** della libreria M258 PLCSystem.

Sono disponibili due **tipi di dati**:

- I **tipi di dati delle variabili di sistema** sono usati dalle **variabili di sistema** (vedi pagina 11) della libreria M258 PLCSystem (PLC_R, PLC_W,...).
- I **tipi di dati delle funzioni di sistema** sono usati dalle **funzioni di sistema** (vedi pagina 33) di lettura/scrittura della libreria M258 PLCSystem.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
3.1	Tipi di dati delle variabili di sistema PLC_R/W	48
3.2	Tipi di dati delle variabili di sistema ETH_R/W	57
3.3	Tipi di dati delle variabili di sistema TM5_MODULE_R/W	65
3.4	Tipi di dati delle funzioni di sistema	66

3.1 Tipi di dati delle variabili di sistema PLC_R/W

Panoramica

In questa sezione sono elencati e descritti i **tipi di dati delle variabili di sistema** inclusi nelle strutture `PLC_R` e `PLC_W`.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
PLC_R_APPLICATION_ERROR: codici di stato degli errori applicazione rilevati	49
PLC_R_BOOT_PROJECT_STATUS: codici di stato del progetto di avvio	50
PLC_R_IO_STATUS: Codici di stato degli I/O	51
PLC_R_STATUS: Codici di stato del controller	52
PLC_R_STOP_CAUSE: codici causa transizione da RUN a un altro stato	53
PLC_R_TERMINAL_PORT_STATUS: codici dello stato di connessione della porta di programmazione	54
PLC_R_USB_HOST_STATUS: codici dello stato di connessione della porta host USB	55
PLC_W_COMMAND: codici dei comandi di controllo	56

PLC_R_APPLICATION_ERROR: codici di stato degli errori applicazione rilevati**Descrizione tipo enumerato**

Il tipo di dati di enumerazione PLC_R_APPLICATION_ERROR contiene i seguenti valori:

Enumeratore	Valore	Commento
PLC_R_APP_ERR_UNKNOWN	FFFF hex	Errore sconosciuto.
PLC_R_APP_ERR_NOEXCEPTION	0000 hex	Nessun errore rilevato.
PLC_R_APP_ERR_WATCHDOG	0010 hex	Watchdog applicazione del task scaduto.
PLC_R_APP_ERR_HARDWAREWATCHDOG	0011 hex	Watchdog hardware scaduto.
PLC_R_APP_ERR_IO_CONFIG_ERROR	0012 hex	Parametri di configurazione I/O errati rilevati.
PLC_R_APP_ERR_UNRESOLVED_EXTREFS	0018 hex	Funzioni sconosciute rilevate.
PLC_R_APP_ERR_IEC_TASK_CONFIG_ERROR	0025 hex	Parametri di configurazione task errati rilevati.
PLC_R_APP_ERR_ILLEGAL_INSTRUCTION	0050 hex	Istruzioni sconosciute rilevate.
PLC_R_APP_ERR_ACCESS_VIOLATION	0051 hex	Accesso all'area di memoria riservata.
PLC_R_APP_ERR_DIVIDE_BY_ZERO	0102 hex	Divisione numero intero per zero rilevata.
PLC_R_APP_ERR_PROCESSORLOAD_WATCHDOG	0105 hex	Processore sovraccarico da task applicazione.
PLC_R_APP_ERR_DIVIDE_REAL_BY_ZERO	0152 hex	Divisione reale per zero rilevata.
PLC_R_APP_ERR_TOO_MANY_EVENT	4E20 hex	La frequenza degli eventi I/O esterni è troppo elevata. Ridurre la frequenza degli eventi Esterni legati ai Task degli eventi.

PLC_R_BOOT_PROJECT_STATUS: codici di stato del progetto di avvio

Descrizione tipo enumerato

Il tipo di dati di enumerazione PLC_R_BOOT_PROJECT_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
PLC_R_NO_BOOT_PROJECT	0000 hex	Progetto di avvio inesistente nella memoria Flash.
PLC_R_BOOT_PROJECT_CREATION_IN_PROGRESS	0001 hex	Creazione del progetto di avvio in corso.
PLC_R_DIFFERENT_BOOT_PROJECT	0002 hex	Il progetto di avvio contenuto nella Flash è diverso dal progetto di avvio caricato nella RAM.
PLC_R_VALID_BOOT_PROJECT	FFFF hex	Il progetto di avvio contenuto nella Flash è identico a quello caricato nella RAM.

PLC_R_IO_STATUS: Codici di stato degli I/O

Descrizione tipo enumerato

Il tipo di dati di enumerazione PLC_R_IO_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
PLC_R_IO_OK	FFFF hex	Ingressi/uscite operativi.
PLC_R_IO_NO_INIT	0001 hex	Ingressi/uscite non inizializzati.
PLC_R_IO_CONF_FAULT	0002 hex	Parametri di configurazione I/O errati rilevati.
PLC_R_IO_SHORTCUT_FAULT	0003 hex	Rilevato cortocircuito degli I/O.
PLC_R_IO_POWER_SUPPLY_FAULT	0004 hex	Errore alimentazione ingressi/uscite rilevato.

PLC_R_STATUS: Codici di stato del controller

Descrizione tipo enumerato

Il tipo di dati di enumerazione PLC_R_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
PLC_R_EMPTY	00 hex	Il controller non è programmato.
PLC_R_STOPPED	01 hex	Il controller è in stop.
PLC_R_RUNNING	02 hex	Il controller è in esecuzione.
PLC_R_HALT	04 hex	Il controller è in uno stato HALT (vedere il Diagramma di stato del controller nella guida alla programmazione del controller).
PLC_R_BREAKPOINT	08 hex	Il controller si mette in pausa al punto di interruzione.

PLC_R_STOP_CAUSE: codici causa transizione da RUN a un altro stato**Descrizione tipo enumerato**

Il tipo di dati di enumerazione PLC_R_STOP_CAUSE contiene i seguenti valori:

Enumeratore	Valore	Commento
PLC_R_STOP_REASON_UNKNOWN	00 hex	Valore iniziale o causa dell'arresto indefinita.
PLC_R_STOP_REASON_HW_WATCHDOG	01 hex	Arresto dopo watchdog hardware.
PLC_R_STOP_REASON_RESET	02 hex	In stop dopo il reset.
PLC_R_STOP_REASON_EXCEPTION	03 hex	In stop dopo un'eccezione.
PLC_R_STOP_REASON_USER	04 hex	Arresto su richiesta dell'utente.
PLC_R_STOP_REASON_IECPROGRAM	05 hex	Arresto dopo richiesta di comando di programma (ad es.: comando di controllo con parametro PLC_W.q_wPLCControl:=PLC_W_COMMAND.PL C_W_STOP;).
PLC_R_STOP_REASON_DELETE	06 hex	In stop dopo un comando di rimozione applicazione.
PLC_R_STOP_REASON_DEBUGGING	07 hex	In stop dopo il passaggio in modalità debug.
PLC_R_STOP_FROM_NETWORK_REQUEST	0A hex	In stop dopo una richiesta dalla rete (chiave USB o comando PLC_W).
PLC_R_STOP_FROM_INPUT	0B hex	Stop richiesto da un ingresso del controller.

PLC_R_TERMINAL_PORT_STATUS: codici dello stato di connessione della porta di programmazione

Descrizione tipo enumerato

Il tipo di dati di enumerazione PLC_R_TERMINAL_PORT_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
TERMINAL_NOT_CONNECTED	00 hex	Nessun PC collegato alla porta di programmazione.
TERMINAL_CONNECTION_IN_PROGRESS	01 hex	Connessione in corso.
TERMINAL_CONNECTED	02 hex	Porta di programmazione collegata a un PC.
TERMINAL_ERROR	0F hex	Errore rilevato durante la connessione.

PLC_R_USB_HOST_STATUS: codici dello stato di connessione della porta host USB

Descrizione tipo enumerato

Il tipo di dati di enumerazione PLC_R_USB_HOST_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
USB_NOT_CONNECTED	00 hex	Nessun dispositivo (chiave di memoria) collegato alla porta USB Host.
USB_CONNECTION_IN_PROGRESS	01 hex	Connessione in corso.
USB_CONNECTED	02 hex	Porta USB Host collegata a un dispositivo (chiave di memoria).
USB_ERROR	0F hex	Errore rilevato durante la connessione.

PLC_W_COMMAND: codici dei comandi di controllo

Descrizione tipo enumerato

Il tipo di dati di enumerazione PLC_W_COMMAND contiene i seguenti valori:

Enumeratore	Valore	Commento
PLC_W_STOP	01 hex	Comando di arresto del controller.
PLC_W_RUN	02 hex	Comando di esecuzione del controller.
PLC_W_RESET_COLD	04 hex	Comando di avvio a freddo del controller.
PLC_W_RESET_WARM	08 hex	Comando di avvio a caldo del controller.

3.2 Tipi di dati delle variabili di sistema ETH_R/W

Panoramica

In questa sezione sono elencati e descritti i **Tipi di dati di sistema** inclusi nelle strutture `ETH_R` e `ETH_W`.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
ETH_R_IP_MODE: codici sorgente indirizzo IP	58
ETH_R_FRAME_PROTOCOL: codici del protocollo di trasmissione dei frame	59
ETH_R_PORT_DUPLEX_STATUS: codici della modalità di trasmissione	60
ETH_R_PORT_LINK_STATUS: codici relativi alla direzione del collegamento di comunicazione	61
ETH_R_PORT_SPEED: codici relativi alla velocità di comunicazione della porta Ethernet	62
ETH_R_PORT_IP_STATUS: Codici di stato porta Ethernet TCP/IP	63
ETH_R_RUN_IDLE: Codici di stato di inattività e funzionamento di Ethernet/IP	64

ETH_R_IP_MODE: codici sorgente indirizzo IP

Descrizione tipo enumerato

Il tipo di dati di enumerazione `ETH_R_IP_MODE` contiene i seguenti valori:

Enumeratore	Valore	Commento
<code>ETH_R_STORED</code>	00 hex	Viene utilizzato l'indirizzo IP memorizzato.
<code>ETH_R_BOOTP</code>	01 hex	Viene utilizzato il protocollo Bootstrap per ottenere un indirizzo IP.
<code>ETH_R_DHCP</code>	02 hex	Viene utilizzato il protocollo DHCP per ottenere un indirizzo IP.
<code>ETH_DEFAULT_IP</code>	FF hex	Viene utilizzato l'indirizzo IP predefinito.

ETH_R_FRAME_PROTOCOL: codici del protocollo di trasmissione dei frame

Descrizione tipo enumerato

Il tipo di dati di enumerazione ETH_R_FRAME_PROTOCOL contiene i seguenti valori:

Enumeratore	Valore	Commento
ETH_R_802_3	00 hex	Il protocollo utilizzato per la trasmissione dei frame è 802.3.
ETH_R_ETHERNET_II	01 hex	Il protocollo utilizzato per la trasmissione dei frame è Ethernet II.

ETH_R_PORT_DUPLEX_STATUS: codici della modalità di trasmissione

Descrizione tipo enumerato

Il tipo di dati di enumerazione ETH_R_PORT_DUPLEX_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
ETH_R_PORT_HALF_DUPLEX	00 hex	Viene utilizzata la modalità di trasmissione half duplex.
ETH_R_FULL_DUPLEX	01 hex	Viene utilizzata la modalità di trasmissione full duplex.

ETH_R_PORT_LINK_STATUS: codici relativi alla direzione del collegamento di comunicazione

Descrizione tipo enumerato

Il tipo di dati di enumerazione ETH_R_PORT_LINK_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
ETH_R_LINK_DOWN	00 hex	Link di comunicazione dal server al dispositivo.
ETH_R_LINK_UP	01 hex	Link di comunicazione dal dispositivo al server.

ETH_R_PORT_SPEED: codici relativi alla velocità di comunicazione della porta Ethernet

Descrizione tipo enumerato

Il tipo di dati di enumerazione `ETH_R_PORT_SPEED` contiene i seguenti valori:

Enumeratore	Valore	Commento
<code>ETH_R_SPEED_10_MB</code>	10 dec.	La velocità di rete è 10 megabit al secondo.
<code>ETH_R_100_MB</code>	100 dec.	La velocità di rete è 100 megabit al secondo.

ETH_R_PORT_IP_STATUS: Codici di stato porta Ethernet TCP/IP

Descrizione tipo enumerato

Il tipo di dati di enumerazione ETH_R_PORT_IP_STATUS contiene i seguenti valori:

Enumeratore	Valore	Commento
WAIT_FOR_PARAMS	00 hex	In attesa dei parametri.
WAIT_FOR_CONF	01 hex	In attesa della configurazione.
DATA_EXCHANGE	02 hex	Pronto per scambio dati.
ETH_ERROR	03 hex	Rilevato errore porta Ethernet TCP/IP (cavo scollegato, configurazione non valida...).
DUPLICATE_IP	04 hex	Indirizzo IP già in uso da un'altra apparecchiatura.

ETH_R_RUN_IDLE: Codici di stato di inattività e funzionamento di Ethernet/IP

Descrizione tipo enumerato

Il tipo di dati di enumerazione `ETH_R_RUN_IDLE` contiene i seguenti valori:

Enumeratore	Valore	Commento
IN ATTESA	00 hex	La connessione EtherNet/IP è inattiva.
RUN	01 hex	La connessione EtherNet/IP è in funzione.

3.3 Tipi di dati delle variabili di sistema

TM5_MODULE_R/W

TM5_MODULE_STATE: codici di stato dei moduli di espansione TM5

Descrizione tipo enumerato

Il tipo di dati di enumerazione TM5_MODULE_STATE contiene i seguenti valori:

Enumeratore	Valore	Commento
TM5_INACTIVE	00 hex	Macchina di stato inattiva.
TM5_BOOT	0A hex	Avvio in corso.
TM5_FWDNLD	0B hex	Download del firmware in corso.
TM5_PREOP	14 hex	Inizializzazione di base.
TM5_OPERATE	1E hex	Inizializzazione registro.
TM5_ACTIVE	64 hex	Comunicazione modulo attiva.
TM5_ERROR	C8 hex	Modulo in stato di errore rilevato.
TM5_UNSUP	C9 hex	Modulo TM5 non supportato.
TM5_NOCFG	CA hex	Nessuna configurazione disponibile.

3.4 Tipi di dati delle funzioni di sistema

Panoramica

Questa sezione descrive i **tipi di dati delle funzioni di sistema** della libreria M258 PLCSystem.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
LED_ID: SetLEDBehaviour Funzione LedId - Codici dei parametri	67
LED_BHV: SetLEDBehaviour Funzione LedBhv - Codici dei parametri	68
LED_BHV_ERROR: Codici errori rilevati della funzione SetLEDBehaviour	69
LED_COLOR: SetLEDBehaviour Funzione LedColor - Codici dei parametri	70
RTCSETDRIFT_ERROR: Codici errori rilevati della funzione SetRTCDrift	71
DAY_OF_WEEK: Codici dei parametri giorno della funzione SetRTCDrift	72
ORA: Tipo di parametro ora della funzione SetRTCDrift	73
MINUTO: Tipo di parametro minuto della funzione SetRTCDrift	74

LED_ID: SetLEDBehaviour Funzione LedId - Codici dei parametri**Descrizione tipo enumerato**

Il tipo di dati di enumerazione LED_ID contiene i seguenti valori:

Enumeratore	Valore	Commento
LED_0	00 hex	Identificativo del LED applicazione APP0.
LED_1	01 hex	Identificativo del LED applicazione APP1.

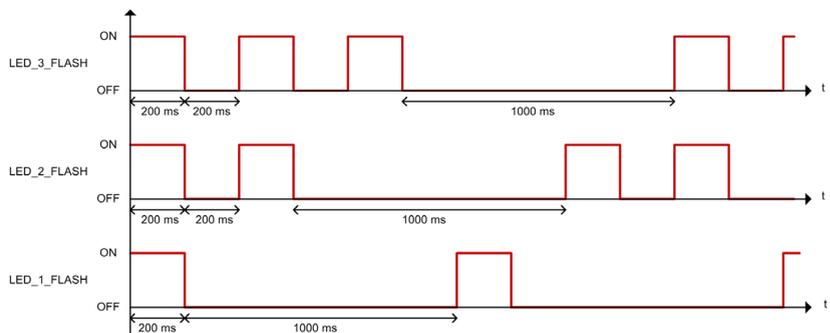
LED_BHV: SetLEDBehaviour Funzione LedBhv - Codici dei parametri

Descrizione tipo enumerato

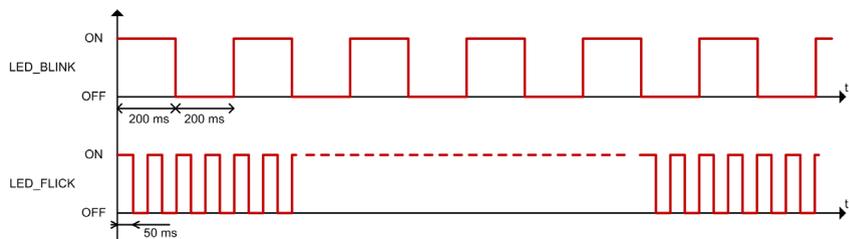
Il tipo di dati di enumerazione LED_BHV contiene i seguenti valori:

Enumeratore	Valore	Commento
LED_3_FLASH	-3 dec.	Il LED lampeggia in modalità di lampeggio 3 (vedere la figura sotto).
LED_2_FLASH	-2 dec.	Il LED lampeggia in modalità di lampeggio 2 (vedere la figura sotto).
LED_1_FLASH	-1 dec.	Il LED lampeggia in modalità di lampeggio 1 (vedere la figura sotto).
LED_OFF	0 dec.	Il LED è sempre spento.
LED_ON	1 dec.	Il LED è sempre acceso.
LED_BLINK	2 dec.	Il LED lampeggia a 2,5 Hz (vedere la figura sotto).
LED_FLICK	3 dec.	Il LED lampeggia a 10 Hz (vedere la figura sotto).

Il seguente grafico illustra le modalità di lampeggio del LED applicazione LED_x_FLASH:



Il seguente grafico illustra le modalità di lampeggio del LED applicazione LED_BLINK and LED_FLICK:



LED_BHV_ERROR: Codici errori rilevati della funzione SetLEDBehaviour**Descrizione tipo enumerato**

Il tipo di dati di enumerazione LED_BHV_ERROR contiene i seguenti valori:

Enumeratore	Valore	Commento
NO_ERROR	00 hex	Funzione di impostazione del comportamento LED eseguita senza errori.
UNKNOWN_LED	01 hex	Parametro LED_ID sconosciuto.
UNKNOWN_COLOR	02 hex	Parametro LED_COLOR sconosciuto.
UNKNOWN_STATE	03 hex	Lo stato dei LED contenuto nel parametro LED_BHV è sconosciuto.
FIRMWARE_ERROR	04 hex	Comando rifiutato dal FW su errore rilevato.

LED_COLOR: SetLEDBehaviour Funzione LedColor - Codici dei parametri

Descrizione tipo enumerato

Il tipo di dati di enumerazione LED_COLOR contiene i seguenti valori:

Enumeratore	Valore	Commento
LED_RED	00 hex	Il LED è rosso.
LED_GREEN	01 hex	Il LED è verde.

RTCSETDRIFT_ERROR: Codici errori rilevati della funzione SetRTCDrift**Descrizione tipo enumerato**

Il tipo di dati di enumerazione RTCSETDRIFT_ERROR contiene i seguenti valori:

Enumeratore	Valore	Commento
RTC_OK	00 hex	deviazione RTC configurata correttamente.
RTC_BAD_DAY	01 hex	Parametro giorno sconosciuto.
RTC_BAD_HOUR	02 hex	Parametro ora sconosciuto.
RTC_BAD_MINUTE	03 hex	Parametro minuto sconosciuto.
RTC_BAD_DRIFT	04 hex	Parametro deviazione RTC fuori intervallo.
RTC_INTERNAL_ERROR	05 hex	Impostazioni deviazione RTC rifiutate su errore interno rilevato.

DAY_OF_WEEK: Codici dei parametri giorno della funzione SetRTCDrift

Descrizione tipo enumerato

Il tipo di dati di enumerazione contiene i seguenti valori:

Enumeratore	Valore	Commento
MONDAY	01 hex	Imposta il giorno della settimana a lunedì
TUESDAY	02 hex	Imposta il giorno della settimana a martedì
WEDNESDAY	03 hex	Imposta il giorno della settimana a mercoledì
THURSDAY	04 hex	Imposta il giorno della settimana a giovedì
FRIDAY	05 hex	Imposta il giorno della settimana a venerdì
SATURDAY	06 hex	Imposta il giorno della settimana a sabato
SUNDAY	07 hex	Imposta il giorno della settimana a domenica

ORA: Tipo di parametro ora della funzione SetRTCDrift

Descrizione del tipo di dati

Questo tipo di dati contiene i valori dell'ora da 0 a 23.

MINUTO: Tipo di parametro minuto della funzione SetRTCDrift

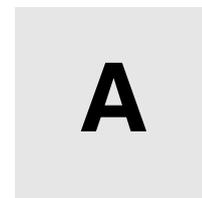
Descrizione del tipo di dati

Questo tipo di dati contiene i valori dei minuti da 0 a 59.

Appendici



Rappresentazione di funzioni e blocchi funzione



Panoramica

Ogni funzione può essere rappresentata nei seguenti linguaggi:

- IL: Instruction List
- ST: Structured Text
- LD: Ladder Diagram
- FBD: Diagramma blocchi funzione
- CFC: Continuous Function Chart

Questo capitolo fornisce esempi di rappresentazioni delle funzioni e dei blocchi funzione e spiega come utilizzarli per i linguaggi IL e ST.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Differenze tra una funzione e un blocco funzione	78
Come utilizzare una funzione o un blocco funzione in linguaggio IL	79
Come utilizzare una funzione o un blocco funzione in linguaggio ST	82

Differenze tra una funzione e un blocco funzione

Funzione

Una funzione:

- è una **POU** (Program Organization Unit) che restituisce un risultato immediato
- è richiamata direttamente con il proprio nome (e non tramite una **Istanza**)
- non ha uno stato persistente da una chiamata all'altra
- può essere utilizzata come operando in altre espressioni

Esempi: operatori booleani (AND), calcoli, conversione (BYTE_TO_INT)

Blocco funzione

Un blocco funzione:

- è una **POU** (Program Organization Unit) che restituisce una o più uscite
- è sempre richiamata tramite una **Istanza** (copia blocco funzione con nome e variabili dedicate)
- ciascuna **Istanza** ha uno stato persistente (uscite e variabili interne) da una chiamata all'altra

Esempi: timer, contatori

Nell'esempio di seguito, `Timer_ON` è un'istanza del blocco funzione `TON`:

```
1  PROGRAM MyProgram_ST
2  VAR
3      Timer_ON: TON; // Function Block Instance
4      Timer_RunCd: BOOL;
5      Timer_PresetValue: TIME := T#5S;
6      Timer_Output: BOOL;
7      Timer_ElapsedTime: TIME;
8  END_VAR
```

```
1  Timer_ON(
2      IN:=Timer_RunCd,
3      PT:=Timer_PresetValue,
4      Q=>Timer_Output,
5      ET=>Timer_ElapsedTime);
```

Come utilizzare una funzione o un blocco funzione in linguaggio IL

Informazioni generali

Questa sezione spiega come implementare una funzione e un blocco funzione in linguaggio IL.

Come esempi, vengono utilizzati le funzioni `IsFirstMastCycle` e `SetRTCDrift` e il blocco funzione `TON`.

Utilizzo di una funzione in linguaggio IL

La procedura descritta spiega come inserire una funzione in linguaggio IL:

Fase	Operazione
1	Aprire o creare una nuova POU in linguaggio Lista d'istruzioni . NOTA: In questa sezione non viene descritta la procedura di creazione di una POU. Per maggiori informazioni, vedere la guida globale di SoMachine.
2	Creare le variabili richieste dalla funzione.
3	Se la funzione ha 1 o più ingressi, iniziare caricando il primo ingresso con l'istruzione LD.
4	Inserire una nuova riga sotto, quindi: <ul style="list-style-type: none"> ● digitare il nome della funzione nella colonna dell'operatore (campo a sinistra), oppure ● utilizzare l'assistente di immissione per selezionare la funzione (selezionare Inserisci modulo nel menu contesto).
5	Se la funzione ha più di 1 ingresso e quando viene utilizzato l' assistente di immissione , il numero necessario di linee viene creato automaticamente con ??? nei campi a destra. Sostituire ??? con il valore appropriato o la variabile che corrisponde all'ordine degli ingressi.
6	Inserire una nuova linea per memorizzare il risultato della funzione nella variabile appropriata: digitare l'istruzione ST nella colonna dell'operatore (campo a sinistra) e il nome di variabile nel campo a destra.

Per illustrare la procedura, considerare le Funzioni `IsFirstMastCycle` (senza parametro di ingresso) e `SetRTCDrift` (con parametri di ingresso) presentate graficamente di seguito:

Funzione	Rappresentazione grafica
senza parametro di ingresso: <code>IsFirstMastCycle</code>	
con parametri di ingresso: <code>SetRTCDrift</code>	

In linguaggio IL, il nome della funzione viene utilizzato direttamente nella **Colonna Operatore**:

Funzione	Rappresentazione nell'editor IL POU di SoMachine
<p>Esempio IL di una funzione senza parametro di ingresso: IsFirstMastCycle</p>	<pre> 1 PROGRAM MyProgram_IL 2 VAR 3 FirstCycle: BOOL; 4 END_VAR 5 </pre> <hr/> <pre> 1 IsFirstMastCycle ST FirstCycle </pre>
<p>Esempio IL di una funzione con parametri di ingresso: SetRTCDrift</p>	<pre> 1 PROGRAM MyProgram_IL 2 VAR 3 myDrift: SINT (-29..29) := 5; 4 myDay: DAY_OF_WEEK := SUNDAY; 5 myHour: HOUR := 12; 6 myMinute: MINUTE; 7 myDiag: RTCSETDRIFT_ERROR; 8 END_VAR 9 </pre> <hr/> <pre> 1 LD myDrift SetRTCDrift myDay myHour myMinute ST myDiag </pre>

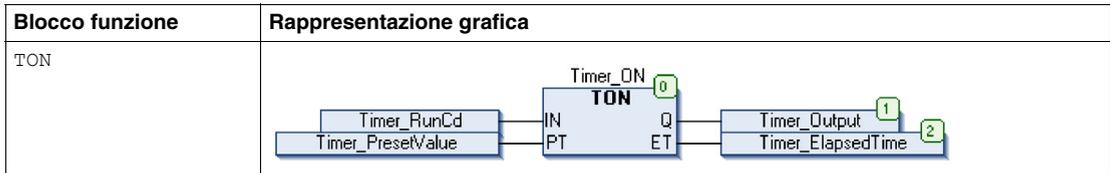
Utilizzo di un blocco funzione in linguaggio IL

La procedura seguente descrive come inserire un blocco funzione in linguaggio IL:

Fase	Operazione
1	<p>Aprire o creare una nuova POU in linguaggio Lista d'istruzioni. NOTA: In questa sezione non viene descritta la procedura di creazione di una POU. Per maggiori informazioni, vedere la guida globale di SoMachine.</p>
2	<p>Creare le variabili richieste dal blocco funzione, incluso il nome dell'istanza.</p>

Fase	Operazione
3	<p>I blocchi funzione vengono chiamati tramite un'istruzione CAL:</p> <ul style="list-style-type: none"> ● utilizzare l'assistente di immissione per selezionare FB (fare clic con il tasto destro e selezionare Inserisci modulo nel menu contesto). ● L'istruzione CAL e gli I/O necessari vengono creati automaticamente. <p>Ciascun parametro (I/O) è un'istruzione:</p> <ul style="list-style-type: none"> ● I valori per gli ingressi vengono impostati da " := ". ● I valori per le uscite vengono impostati da " => ".
4	Nel campo sul lato destro CAL, sostituire ??? con il nome dell'istanza.
5	Sostituire altri ??? con una variabile appropriata o un valore immediato.

Per illustrare la procedura, considerare questo esempio con il blocco funzione TON rappresentato graficamente di seguito:



In linguaggio IL, il nome del blocco funzione viene utilizzato direttamente nella **Colonna Operatore**:

Blocco funzione	Rappresentazione nell'editor IL POU di SoMachine
TON	<pre> 1 PROGRAM MyProgram_IL 2 VAR 3 Timer_ON: TON; // Function Block instance declaration 4 Timer_RunCd: BOOL; 5 Timer_PresetValue: TIME := T#5S; 6 Timer_Output: BOOL; 7 Timer_ElapsedTime: TIME; 8 END_VAR 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 </pre>

Come utilizzare una funzione o un blocco funzione in linguaggio ST

Informazioni generali

Questa sezione spiega come implementare una funzione e un blocco funzione nel linguaggio ST.

Come esempi vengono utilizzati la funzione `SetRTCDrift` e il blocco funzione `TON`.

Utilizzo di una funzione in linguaggio ST

La procedura descritta spiega come inserire una funzione in linguaggio ST:

Fase	Operazione
1	Aprire o creare una nuova POU nel linguaggio Testo strutturato . NOTA: In questa sezione non viene descritta la procedura di creazione di una POU. Per maggiori informazioni, vedere la guida globale di SoMachine.
2	Creare le variabili richieste dalla funzione.
3	Utilizzare la sintassi generale nell' editor ST POU per il linguaggio ST di una funzione. La sintassi generale è: <code>FunctionResult:= FunctionName(VarInput1, VarInput2,.. VarInputx);</code>

Per illustrare la procedura, considerare la funzione `SetRTCDrift` rappresentata graficamente di seguito:



Il linguaggio ST di questa funzione è il seguente:

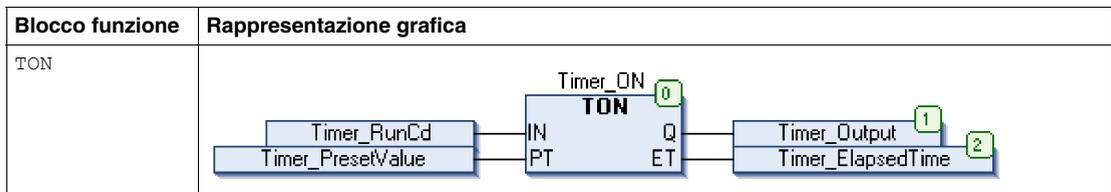
Funzione	Rappresentazione nell'editor ST POU di SoMachine
SetRTCDrift	<pre>PROGRAM MyProgram_ST VAR myDrift: SINT(-29..29) := 5; myDay: DAY_OF_WEEK := SUNDAY; myHour: HOUR := 12; myMinute: MINUTE; myRTCAdjust: RTCDRIFT_ERROR; END_VAR myRTCAdjust:= SetRTCDrift(myDrift, myDay, myHour, myMinute);</pre>

Utilizzo di un blocco funzione in linguaggio ST

La procedura seguente descrive come inserire un blocco funzione in linguaggio ST:

Fase	Operazione
1	<p>Aprire o creare una nuova POU nel linguaggio Testo strutturato.</p> <p>NOTA: In questa sezione non viene descritta la procedura di creazione di una POU. Per maggiori informazioni, vedere la guida globale di SoMachine.</p>
2	<p>Creare le variabili di ingresso e uscita e l'istanza richiesta per il blocco funzione:</p> <ul style="list-style-type: none"> Le variabili di ingresso sono i parametri di ingresso richiesti dal blocco funzione Alle variabili di uscita viene assegnato il valore restituito dal blocco funzione
3	<p>Utilizzare la sintassi generale nell'editor ST POU per il linguaggio ST di un blocco funzione. La sintassi generale è:</p> <pre>FunctionBlock_InstanceName (Input1:=VarInput1, Input2:=VarInput2,... Ouput1=>VarOutput1, Ouput2=>VarOutput2,...);</pre>

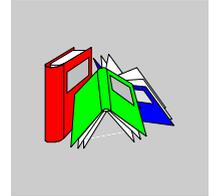
Per illustrare la procedura, considerare questo esempio con il blocco funzione TON rappresentato graficamente di seguito:



Nella tabella seguente sono riportati esempi di chiamata di un blocco funzione nel linguaggio ST:

Blocco funzione	Rappresentazione nell'editor ST POU di SoMachine
TON	<pre>1 PROGRAM MyProgram_ST 2 VAR 3 Timer_ON: TON; // Function Block Instance 4 Timer_RunCd: BOOL; 5 Timer_PresetValue: TIME := T#5S; 6 Timer_Output: BOOL; 7 Timer_ElapsedTime: TIME; 8 END_VAR 1 Timer_ON(2 IN:=Timer_RunCd, 3 PT:=Timer_PresetValue, 4 Q=>Timer_Output, 5 ET=>Timer_ElapsedTime);</pre>

Glossario



A

applicazione di avvio

File che contengono parametri dipendenti dalla macchina:

- nome macchina
- nome dispositivo o indirizzo IP
- indirizzo linea seriale Modbus
- tabella di instradamento

B

Blocco funzione

(FB). Un'unità di programma di ingressi e variabili che permette di calcolare i valori delle uscite in base a una funzione definita, come un timer o un contatore.

BOOL

Un tipo *booleano* è il tipo di dati di base in informatica. Una variabile `BOOL` può avere uno di questi valori: 0 (`FALSE`), 1 (`TRUE`). Un bit estratto da una parola è di tipo `BOOL`, ad esempio: `%MW10.4` è un quinto bit del numero della parola di memoria 10.

BOOTP

Il *Bootstrap Protocol* è un protocollo di rete UDP che può essere utilizzato da un client di rete per ottenere automaticamente un indirizzo IP (ed eventualmente altri dati) da un server. Il client si identifica per il server con l'indirizzo MAC del client. Il server, che mantiene una tabella preconfigurata di indirizzi MAC di dispositivi client e relativi indirizzi IP, invia al client il suo indirizzo IP preconfigurato. Originariamente, BOOTP era utilizzato come metodo per consentire l'avvio remoto di host senza disco tramite rete. Il processo BOOTP assegna gli indirizzi IP per un periodo di tempo indefinito. Il servizio BOOTP utilizza le porte UDP 67 e 68.

bus di espansione

Il *bus di espansione* è un bus di comunicazione elettronico tra i moduli di espansione e una CPU.

BYTE

Un gruppo di 8 bit è definito **BYTE**. Si può immettere un **BYTE** in modalità binaria o in base 8. Il tipo **BYTE** è codificato in formato a 8 bit che va da 16#00 a 16#FF (in formato esadecimale).

C

CAN

Il protocollo *Controller Area Network* (ISO 11898), per le reti di bus seriali è stato progettato per l'interconnessione di dispositivi smart (di vari costruttori) in sistemi smart per applicazioni industriali in tempo reale. I sistemi CAN multi-master assicurano l'integrità dei dati attraverso l'implementazione di messaggia broadcast e di meccanismi diagnostici avanzati. Originariamente sviluppato per l'industria automobilistica, CAN è ora utilizzato in molte applicazioni per il controllo dei processi di automazione industriali.

CANopen

CANopen è un protocollo di comunicazione aperto di standard industriale e una specifica del profilo dispositivo.

CFC

Il *linguaggio CFC (Continuous Function Chart)*, un'estensione dello standard IEC61131-3, è un linguaggio di programmazione grafica strutturato come un diagramma di flusso. Ogni funzione o blocco funzione del programma viene rappresentato in questo formato grafico combinando blocchi logici semplici (AND, OR, ecc.). Per ogni blocco, gli ingressi si trovano a sinistra e le uscite a destra. È possibile collegare le uscite dei blocchi agli ingressi di altri blocchi per formare espressioni complesse.

contatore 1-Phase

Il *contatore 1-Phase* utilizza un ingresso hardware come ingresso contatore. In genere effettua il conteggio avanti o indietro quando è presente un segnale d'impulsi sull'ingresso.

contatore 2-Phase

Il *contatore 2-Phase* utilizza la differenza di fase tra due segnali del contatore d'ingresso per contare avanti o indietro.

controller

Un *controller*, o PLC (Programmable Logic Controller), viene utilizzato per automatizzare i processi industriali.

D**dati ritenuti**

Un valore di *dato ritenuto* è un valore che viene utilizzato all'accensione o al riavvio a caldo successivi. Questo valore viene mantenuto anche dopo un'interruzione imprevista dell'alimentazione del controller o lo spegnimento normale del controller.

F**FBD**

Un *diagramma blocco funzione* è un linguaggio di programmazione a grafici, conforme allo standard IEC 61131-3. È basato su una serie di reti, ognuna delle quali contiene una struttura grafica composta da quadrati e linee di collegamento che rappresentano un'espressione logica o aritmetica, il richiamo di un blocco funzione, un salto o un'istruzione di ritorno.

firmware

Il *firmware* rappresenta il sistema operativo di un controller.

G

GVL

L'elenco *GVL (Global Variable List)* gestisce le variabili globali disponibili in ogni POU dell'applicazione.

H

HMI

Un'*HMI (Human-Machine Interface, Interfaccia uomo-macchina)* è un'interfaccia operatore, in genere grafica, per le apparecchiature di uso industriale.

HSC

contatore ad alta velocità

I

IEC 61131-3

Lo standard IEC 61131-3 è una norma della *commissione elettrotecnica internazionale* relativa ai dispositivi di automazione industriale (come i controller). Lo standard IEC 61131-3 riguarda i linguaggi di programmazione dei controller e definisce due normative per i linguaggi di programmazione grafici e due per i linguaggi testuali:

- **linguaggi grafici:** diagramma ladder, diagramma blocchi funzione
- **linguaggi testuali:** testo strutturato, lista di istruzioni

IL

Un programma scritto nel linguaggio *Lista di istruzioni* è composto da una serie di istruzioni eseguite in modo sequenziale dal controller. Ogni istruzione è composta da un numero di riga, un codice istruzione e un operando. (Il linguaggio IL è conforme alla norma IEC 61131-3).

L

LD

Un programma scritto in linguaggio a schema *ladder* (*LD*), ossia una rappresentazione grafica delle istruzioni di un programma del controller con simboli che rappresentano i contatti, le bobine e i blocchi funzione in una serie di reti (*rung*) eseguite in modo sequenziale da un controller. Questo linguaggio è conforme allo standard IEC 61131-3.

linguaggio Diagramma blocchi funzione

(FBD). Un diagramma blocco funzione descrive una funzione tra le variabili di ingresso e le variabili di uscita. Descrive una funzione come insieme di blocchi elementari. Le variabili di ingresso e di uscita sono collegate ai blocchi mediante linee di collegamento. Un'uscita di un blocco può anche essere collegata all'ingresso di un altro blocco.

M

Modbus

Il protocollo di comunicazione Modbus permette la comunicazione tra più dispositivi collegati alla stessa rete.

modulo d'espansione degli I/O

Un *modulo di espansione degli ingressi o delle uscite* è un modulo digitale o analogico che aggiunge degli I/O al controller di base.

N

NEMA

NEMA (National Electrical Manufacturers Association) è l'ente preposto alla pubblicazione degli standard relativi alle caratteristiche di cabinet elettrici di diverse classi. Gli standard NEMA riguardano la resistenza alla corrosione, la capacità di protezione dalla pioggia e dall'immersione, ecc. Per gli stati la cui legislazione aderisce alle normative IEC, lo standard IEC 60529 classifica il grado di tenuta dei cabinet.

P

PLC

Il *PLC* è il centro di elaborazione di un processo di produzione industriale. Utilizzato in sostituzione dei sistemi controllati da relè, effettua l'automatizzazione del processo. I PLC sono computer previsti per operare nelle condizioni critiche tipiche degli ambienti industriali.

PLI

Ingresso latch a impulsi

POU

Un'*unità di organizzazione dei programmi* che include una dichiarazione di variabili nel codice sorgente e il set di istruzioni corrispondente. Le POU semplificano il riutilizzo modulare di programmi software, funzioni e blocchi funzione. Una volta dichiarate, le POU sono reciprocamente disponibili. La programmazione di SoMachine richiede l'utilizzo delle POU.

PTO

Le *uscite a treno di impulsi (Pulse Train Outputs)* permettono di controllare, ad esempio, i motori passo passo in anello aperto.

PWM

La *modulazione ad ampiezza di impulsi (Pulse Width Modulation)* viene utilizzata per i processi di regolazione (ad esempio gli attuatori per il controllo della temperatura) nei quali un segnale impulso viene modulato nella sua lunghezza. Per questo tipo di segnali vengono utilizzate le uscite transistor.

R

RTC

L'opzione *orologio in tempo reale (RTC, real-time clock)* è un'opzione che permette di mantenere l'indicazione oraria per un periodo di tempo limitato quando il controller non è sotto tensione.

S

SFC

Un programma scritto in un linguaggio *grafico di funzione sequenziale* o SFC può essere utilizzato per i processi che possono essere suddivisi in fasi. Il linguaggio SFC è costituito da passi a cui sono associate azioni e transizioni logiche con relative condizioni logiche e collegamenti diretti tra passi e transizioni. Lo standard SFC è definito nella norma IEC 848 ed è conforme alla norma IEC 61131-3.

T

Testo strutturato

Un programma scritto in linguaggio di testo strutturato *ST* include istruzioni complesse e istruzioni annidate (come loop di iterazione, esecuzioni condizionali o funzioni). *ST* è conforme allo standard IEC 61131-3.

U

uscita riflessa

In una modalità di conteggio, il valore corrente del contatore ad alta velocità viene misurato rispetto alle soglie configurate per determinare lo stato di queste uscite dedicate.

V

variabile di sistema

Una struttura della variabile di sistema fornisce i dati del controller e informazioni di diagnostica e consente l'invio di comandi al controller.

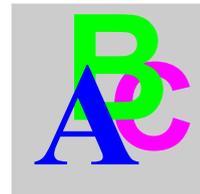
variabile identificata

Una *variabile identificata* ha un indirizzo. (Vedere *variabile non identificata*).

variabile non identificata

Una *variabile non identificata* non ha un indirizzo. (Vedere *variabile identificata*).

Indice analitico



D

DAY_OF_WEEK

Tipi di dati, 72

DM72FGetImmediateInput

Funzioni, 35

E

ETH_R

Variabile di sistema, 27

ETH_R_FRAME_PROTOCOL

Tipi di dati, 59

ETH_R_IP_MODE

Tipi di dati, 58

ETH_R_PORT_DUPLEX_STATUS

Tipi di dati, 60

ETH_R_PORT_LINK_STATUS

Tipi di dati, 61

ETH_R_PORT_SPEED

Tipi di dati, 62

ETH_W

Variabile di sistema, 31

F

Funzioni

Come utilizzare una funzione o un blocco
funzione in linguaggio IL, 79

come utilizzare una funzione o un blocco
funzione in linguaggio ST, 82

Differenze tra una funzione e un blocco

funzione, 78

DM72FGetImmediateInput, 35

getTM5Delay, 36

IsFirstMastColdCycle, 38

IsFirstMastCycle, 39

IsFirstMastWarmCycle, 41

SetLEDBehaviour, 43

SetRTCDrift, 45

G

getTM5Delay

Funzioni, 36

I

IsFirstMastColdCycle

Funzioni, 38

IsFirstMastCycle

Funzioni, 39

IsFirstMastWarmCycle

Funzioni, 41

L

LED_BHV

Tipi di dati, 68

LED_BHV_ERROR

Tipi di dati, 69

LED_COLOR

Tipi di dati, 70

LED_ID

Tipi di dati, 67

M

MINUTO

Tipi di dati, 74

O

ORA

Tipi di dati, 73

P

PLC_R

Variabile di sistema, 18

PLC_R_APPLICATION_ERROR

Tipi di dati, 49

PLC_R_BOOT_PROJECT_STATUS

Tipi di dati, 50

PLC_R_IO_STATUS

Tipi di dati, 51

PLC_R_STATUS

Tipi di dati, 52

PLC_R_STOP_CAUSE

Tipi di dati, 53

PLC_R_TERMINAL_PORT_STATUS

Tipi di dati, 54

PLC_R_USB_HOST_STATUS

Tipi di dati, 55

PLC_W

Variabile di sistema, 22

PLC_W_COMMAND

Tipi di dati, 56

R

RTCSETDRIFT_ERROR

Tipi di dati, 71

S

SERIAL_R

Variabile di sistema, 24

SERIAL_W

Variabile di sistema, 25

SetLEDBehaviour

Funzioni, 43

SetRTCDrift

Funzioni, 45

T

Tipi di dati

DAY_OF_WEEK, 72

ETH_R_FRAME_PROTOCOL, 59

ETH_R_IP_MODE, 58

ETH_R_PORT_DUPLEX_STATUS, 60

ETH_R_PORT_IP_STATUS, 63

ETH_R_PORT_LINK_STATUS, 61

ETH_R_PORT_SPEED, 62

ETH_R_RUN_IDLE, 64

LED_BHV, 68

LED_BHV_ERROR, 69

LED_COLOR, 70

LED_ID, 67

MINUTO, 74

ORA, 73

PLC_R_APPLICATION_ERROR, 49

PLC_R_BOOT_PROJECT_STATUS, 50

PLC_R_IO_STATUS, 51

PLC_R_STATUS, 52

PLC_R_STOP_CAUSE, 53

PLC_R_TERMINAL_PORT_STATUS,
54

PLC_R_USB_HOST_STATUS, 55

PLC_W_COMMAND, 56

RTCSETDRIFT_ERROR, 71

TM5_MODULE_STATE, 65

TM5_MODULE_R

Variabile di sistema, 32

TM5_MODULE_STATE

Tipi di dati, 65

V

Variabile di sistema

- ETH_R, 27
- ETH_W, 31
- PLC_R, 18
- PLC_W, 22
- SERIAL_R, 24
- SERIAL_W, 25
- TM5_MODULE_R, 32

Variabili di sistema

- definizione, 13
- uso, 15

