

MODULO 2

ELEMENTI DI PROGRAMMAZIONE DEL PLC S7-1200 SIEMENS

PARTE PRIMA

IL LINGUAGGIO A CONTATTI

.....

2.1 CONTATTI E BOBINE (KOP).....pag.	2
2.2 ISTRUZIONI DI IMPOSTAZIONE E RESET.....pag.	3
2.3 LATCH DI BIT CON SET E RESET DOMINANTEpag.	5
2.4 ISTRUZIONI DI FRONTE DI SALITA E DI DISCESApag.	6
2.5 TEMPORIZZATORIpag.	8
2.6 CONTATORIpag.	13
2.7 CONFRONTOpag.	16
2.8 TIPI DI DATIpag.	19

Modulo realizzato dal Prof. Salvatore Papotto a supporto della didattica.
Le unità didattiche, tratte dal manuale di sistema, sono state integrate con spiegazioni ed esempi.

2.1 CONTATTI E BOBINE D'USCITA (KOP)

CONTATTI NORMALMENTE APERTI E CHIUSI

E' possibile collegare contatti con altri contatti e realizzare delle combinazioni logiche.
Se il bit d'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.

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.

Il contatto NOT 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

- Se una bobina di uscita è attraversata dal flusso di corrente, il bit di uscita viene impostato a 1.
- Se una 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.



2.2 ISTRUZIONI DI IMPOSTAZIONE E RESET

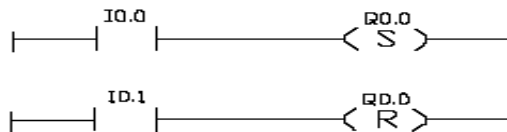
IMPOSTA E RESETTA 1 BIT

- Quando S (imposta) è attivata, il valore di dati nell'indirizzo OUT viene impostato a 1. Quando S è disattivata, OUT resta invariato.
- Quando R (resetta) è attivata, il valore di dati nell'indirizzo OUT viene impostato a 0. Quando R è disattivata, OUT resta invariato.



ESEMPIO

Se si preme il pulsante I0.0 (anche se si rilascia) la lampada di segnalazione si accende.
Per spegnere la lampada bisogna premere il pulsante I0.1.
Q0.0 (lampada di segnalazione)

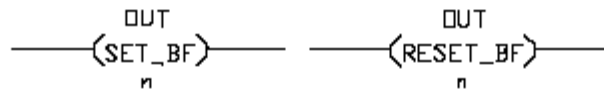


IMPOSTA E RESETTA CAMPO DI BIT

- Quando SET_BF è attivata, viene assegnato il valore di dati 1 a n bit a partire dall'indirizzo OUT.

Quando SET_BF è disattivata, OUT resta invariato.

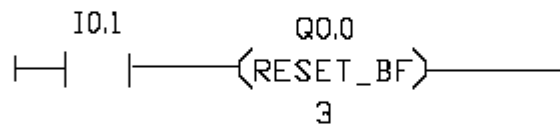
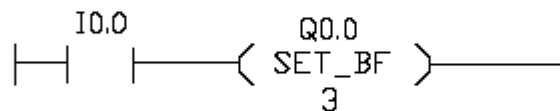
- RESET_BF scrive il valore di dati 0 in n bit a partire dall'indirizzo OUT. Quando RESET_BF è disattivata, OUT resta invariato.



ESEMPIO

Supponiamo di avere tre lampade di segnalazione (Q0.0, Q0.1, Q0.2).

Se si preme il pulsante I0.0 (anche se si rilascia) le tre lampade si accendono; per spegnerle bisogna premere il pulsante I0.1.



2.3 LATCH DI BIT CON SET E RESET DOMINANTE

RS è un latch in cui è dominante set.

Se i segnali d'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 d'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.

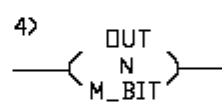
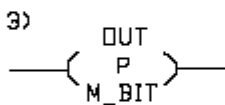
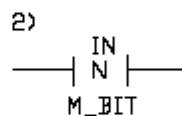
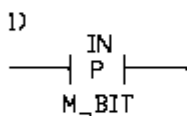
Istruzione: RS	S1	R	Bit "OUT"
	0	0	Stato precedente
	0	1	0
	1	0	1
	1	1	1

Istruzione: SR	S	R1	Bit "OUT"
	0	0	Stato precedente
	0	1	0
	1	0	1
	1	1	0



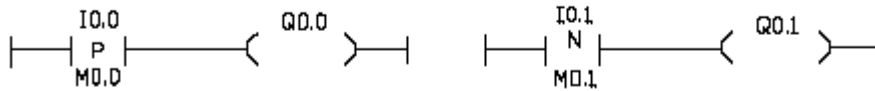
2.4 ISTRUZIONI DI FRONTE DI SALITA E DI DISCESA

- 1) Lo stato di questo contatto è vero quando viene rilevata una transizione positiva (da OFF a ON) nel bit assegnato "IN".
Il contatto P può essere inserito in qualsiasi punto del segmento tranne che alla fine del ramo.
- 2) Lo stato di questo contatto è vero quando viene rilevata una transizione negativa (da ON a OFF) nel bit d'ingresso assegnato.
Il contatto N può essere inserito in qualsiasi punto del segmento tranne che alla fine del ramo.
- 3) Il bit assegnato "OUT" è vero quando viene rilevata una transizione positiva (da OFF a ON) nel flusso di corrente in ingresso alla bobina.
La bobina P può essere inserita in qualsiasi punto del segmento.
- 4) Il bit assegnato "OUT" è vero quando viene rilevata una transizione negativa (da ON a OFF) nel flusso di corrente in ingresso alla bobina.
La bobina N può essere inserita in qualsiasi punto del segmento.
Tutte le istruzioni di fronte utilizzano un merker (M_BIT) per memorizzare lo stato precedente del segnale d'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.



ESEMPIO

Quando si chiude I0.0 (interruttore cablato NA), Q0.0 si attiva per un ciclo di scansione.
Quando si apre I0.1 (interruttore cablato NC), Q0.1 si attiva per un ciclo di scansione.

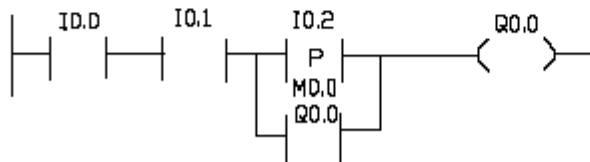


MARCIA E ARRESTO DI UN MOTORE ASINCRONO TRIFASE

Se si comanda un m.a.t. mediante l'istruzione di rilevamento del fronte di salita, nel caso in cui il pulsante di avviamento rimanga chiuso e si preme il pulsante di arresto I0.0 il motore si ferma (perché si diseccita Q0.0), ma al rilascio del pulsante I0.0 il motore non riparte, perché I0.2 non è passato da OFF a ON.

Senza l'istruzione di rilevamento del fronte di salita il motore sarebbe ripartito e la macchina da esso comandata avrebbe potuto causare dei danni.

SCHEMA A CONTATTI



I0.0 (pulsante di arresto cablato NC)

I0.1 (contatto comandato dal relè termico cablato NC)

I0.2 (pulsante di avviamento)

Q0.0 (bobina del contattore che comanda il motore)

MO.0 (marker di memorizzazione dello stato del segnale di ingresso)

2.5

TEMPORIZZATORI

Le istruzioni di temporizzazione sono utilizzate per creare ritardi programmati.

Il numero di temporizzatori utilizzabili nel programma utente è limitato unicamente dalla quantità di memoria disponibile nella CPU.

Ogni temporizzatore utilizza una struttura di DB (blocco dati) del tipo dati IEC_Timer di 16 byte per memorizzare i dati del temporizzatore specificati nella parte superiore del box.

STEP 7 crea automaticamente il DB all'inserimento dell'istruzione.

I valori di PT (tempo preimpostato) ed ET (tempo trascorso) vengono memorizzati nei dati del DB IEC_Timer come numeri interi con segno che rappresentano i millisecondi.

I dati di TIME utilizzano l'ID T# e possono essere specificati come unità di tempo semplice (T#100ms o 100) o composta come T#1m_30s_15ms

TIPI DI DATI PER I PARAMETRI

PARAMETRI	TIPO DI DATI
IN, R, Q	BOOL
PT, ET	TIME

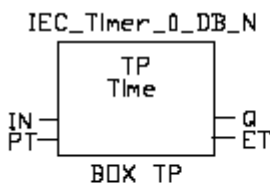
DIMENSIONE E CAMPO DEL TIPO DI DATI TIME

TIPO DI DATI	DIMENSIONE	CAMPI NUMERICI VALIDI
TIME	32 bit, salvati come dati DInt	T#-24d_20h_31m_23s_648ms..... T#24d_20h_31m_23s_647ms Salvati come: -2.147.483.648 ms..... +2.147.483.647 ms

Nota: il campo negativo del tipo di dati TIME indicato in tabella non è utilizzabile con le istruzioni di temporizzazione.

TEMPORIZZATORE TP (AD IMPULSO)

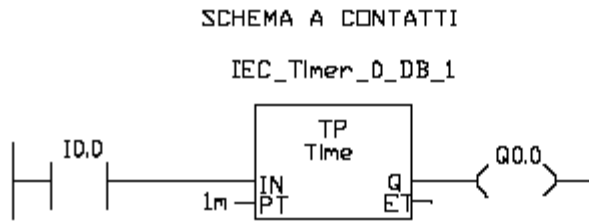
Il temporizzatore TP genera un impulso con una durata preimpostata.



IN<Ingresso di abilitazione del temporizzatore>
PT<Ingresso tempo preimpostato>
Q<uscita del box>
ET<tempo trascorso>
N<numero del temporizzatore: 0,1,2,3.....>

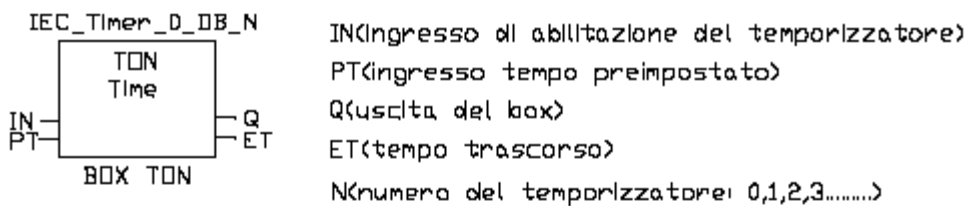
ESEMPIO

Se si preme il pulsante I0.0 (anche se poi si rilascia) la bobina Q0.0 viene attivata per il tempo impostato (un minuto).



TEMPORIZZATORE TON

Il temporizzatore TON imposta l'uscita Q su ON al termine di un tempo di ritardo preimpostato.



ESEMPIO

Comando di un m.a.t. con arresto temporizzato:

I0.0 (pulsante di arresto cablato NC), I0.1 (contatto comandato dal relè termico cablato NC), I0.2 (pulsante di avviamento), Q0.0 (bobina del contattore che comanda il motore).

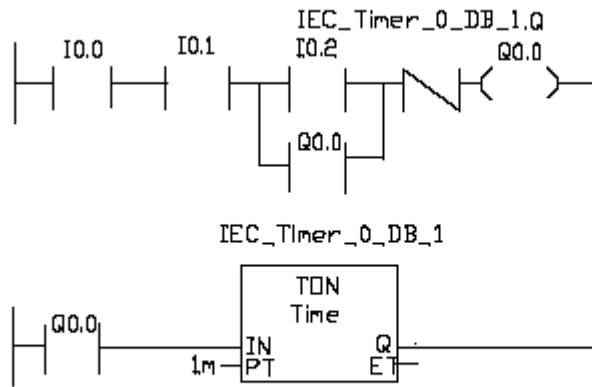
Nota: i contatti con l'indirizzo Q0.0 sono comandati dalla bobina Q0.0.

Funzionamento

Quando si preme il pulsante di avviamento I0.2 si attiva la bobina Q0.0 che chiude il contatto di ritenuta in parallelo al pulsante I0.2 e il contatto Q0.0 della linea di abilitazione del timer.

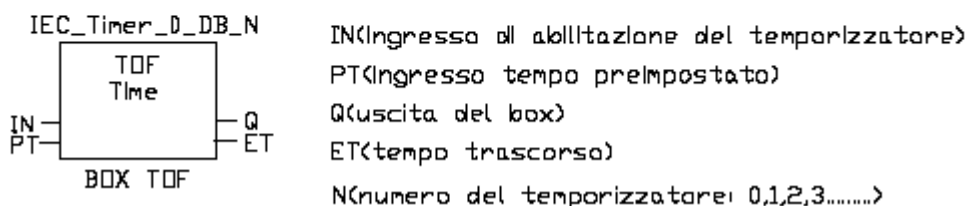
Trascorso il tempo impostato (un minuto), il temporizzatore apre il contatto NC "IEC_Timer_0_DB_1.Q" e il motore si ferma (perché si disattiva la bobina Q0.0).

SCHEMA A CONTATTI



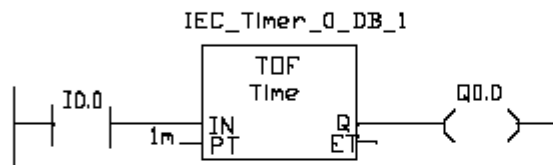
TEMPORIZZATORE TOF

Il temporizzatore TOF resetta l'uscita Q su OFF al termine di un tempo di ritardo preimpostato.



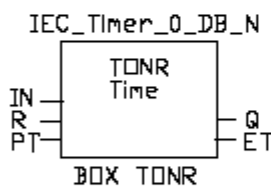
ESEMPIO

Se si preme il pulsante I0.0 la bobina Q0.0 viene attivata, al rilascio del pulsante, trascorso il tempo impostato (un minuto), la bobina viene disattivata.



TEMPORIZZATORE TONR

Il temporizzatore TONR imposta l'uscita Q 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.



IN(Ingresso di abilitazione del temporizzatore)
PT(Ingresso tempo preimpostato)
Q(uscita del box)
ET(tempo trascorso)
R(resetta tempo trascorso)
N(numero del temporizzatore: 0,1,2,3.....)

ESEMPIO

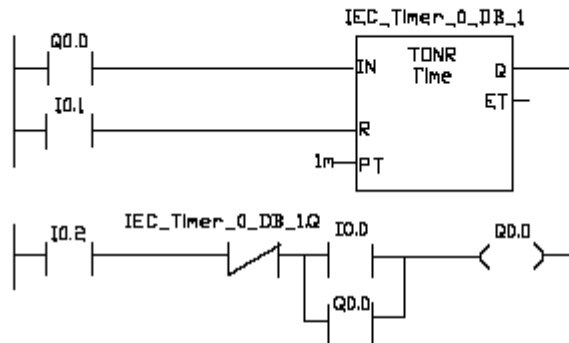
Premendo il pulsante I0.0 si attiva la bobina Q0.0, si chiude il contatto Q0.0 posto nella linea d'ingresso del temporizzatore e il TONR inizia il conteggio.

Se premiamo il pulsante di arresto I0.2 (cablato NC) prima che sia trascorso il tempo impostato, la bobina Q0.0 si disattiva e i contatti Q0.0 si riaprono.

Ripremendo il pulsante I0.0 la bobina Q0.0 si attiva di nuovo e chiude i contatti Q0.0.

Il temporizzatore riprende il conteggio dal punto, dove era stato interrotto e raggiunto il tempo impostato (un minuto) apre il contatto NC "IEC_Timer_0_DB_1".Q disattivando la bobina Q0.0.

Il pulsante I0.1 serve a resettare il tempo trascorso (R=0 nessun reset, R=1 reset).



2.6

CONTATORI

I contatori consentono di contare i vari eventi di un processo.

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.

I contatori disponibili sono: CTU, CTD e CTUD.

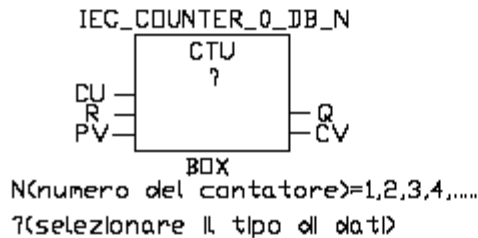
CTU

Il contatore CTU è un contatore con conteggio in avanti.

Il contatore 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.

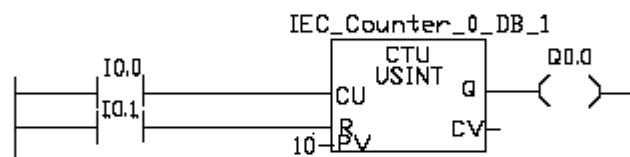
Se il valore del parametro di reset R cambia da 0 a 1, il valore di conteggio attuale viene resettato a 0.



ESEMPIO

Se premiamo dieci volte il pulsante I0.0 il conteggio arriva a 10 e la bobina Q0.0 si attiva.

Se si preme il pulsante I0.1 il conteggio viene resettato e la bobina Q0.0 si disattiva.



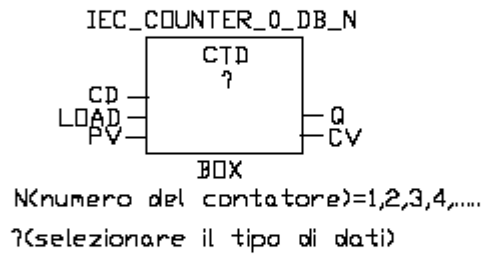
CTD

Il contatore CTD è un contatore con conteggio all'indietro.

Il contatore CTD conta indietro di 1 quando il valore del parametro CD cambia da 0 a 1.

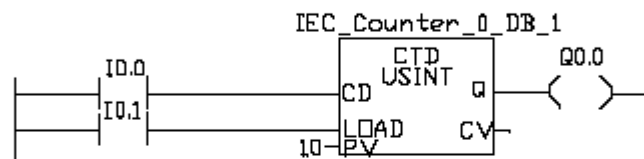
Se il valore del parametro CV (valore di conteggio attuale) è inferiore o uguale a 0, il parametro di uscita del contatore Q = 1.

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).



ESEMPIO

Se premiamo dieci volte il pulsante I0.0 il conteggio arriva a zero e la bobina Q0.0 si attiva.
Se si preme il pulsante I0.1 il valore 10 viene ricaricato nel contatore e la bobina Q0.0 si disattiva.



CTUD

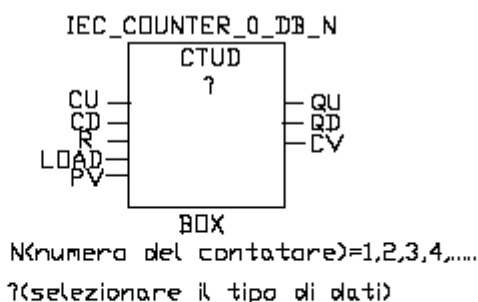
Il CTUD è un contatore con conteggio in avanti e all'indietro.
Il contatore CTUD conta in avanti e all'indietro di 1 quando gli ingressi di conteggio in avanti (CU) o all'indietro (CD) passano da 0 a 1.

Se il valore del parametro CV è maggiore o uguale al valore del parametro PV, il parametro di uscita del contatore QU = 1.

Se il valore del parametro CV è minore 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 viene caricato nel contatore come nuovo CV.

Se il valore del parametro reset R cambia da 0 a 1, il valore di conteggio attuale viene resettato a zero.



ESEMPIO

IL contatore CTUD può essere utilizzato nell'automazione di un parcheggio, per segnalare quando è pieno o ci sono posti liberi.

In questo caso l'ingresso di conteggio in avanti (CU) è un contatto della fotocellula che intercetta le macchine in entrata e l'ingresso di conteggio all'indietro (CD) è un contatto della fotocellula che intercetta le macchine in uscita.

La differenza tra le macchine in entrata e quelle in uscita rappresenta il numero di macchine presenti nel parcheggio ed è fornito dal valore CV.

Confrontando il valore CV con il numero di posti del parcheggio (capienza) s'individua se il parcheggio è pieno o ci sono posti liberi (vedi esempio successivo).

TIPI DI DATI PER I PARAMETRI

Parametro	Tipo di dati	Descrizione
CU, CD	Bool	Conta in avanti o indietro di uno
R (presente in CTU e CTUD)	Bool	Resetta a zero il valore di conteggio
LD (presente in CTD e CTUD)	Bool	Carica il controllo per il valore preimpostato
PV	SInt, Int, DInt, USInt, UInt, UDIInt (vedi tipi di dati)	Valore del conteggio preimpostato
Q, QU	Bool	Vero se $CV \geq PV$
QD	Bool	Vero se $CV \leq 0$
CV	SInt, Int, DInt, USInt, UInt, UDIInt (vedi tipi di dati)	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.

Il numero di contatori utilizzabili nel programma utente è limitato unicamente dalla quantità di memoria disponibile nella CPU.

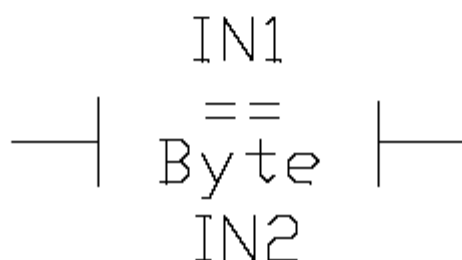
I contatori utilizzano il seguente spazio di memoria:

- 3 byte, per tipi di dati SInt e USInt;
- 6 byte, per i tipi di dati Int e UInt;
- 12 byte, per i tipi di dati DInt e UDInt.

2.7 CONFRONTO

L'istruzione confronta due valori dello stesso tipo di dati.

Se il confronto del contatto KOP è vero, il contatto viene attivato.



CONTATTO DI CONFRONTO

Per selezionare l'istruzione fare clic sul nome (ad es.==).

Per scegliere il tipo di dati fare clic su "???"(ad.es. Byte).

TIPI DI DATI PER I PARAMETRI

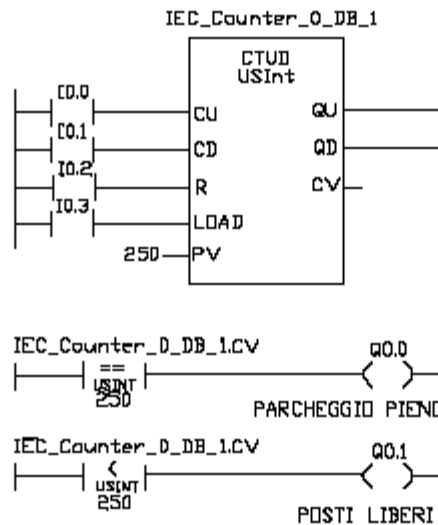
PARAMETRO	TIPO DI DATI	DESCRIZIONE
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, String, Char, Time, DTL, costante. (vedi tipi di dati)	Valori da confrontare

DESCRIZIONE DEL CONFRONTO

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 di IN2

ESEMPIO

SEGNALAZIONE DI PARCHEGGIO AUTO PIENO E POSTI LIBERI

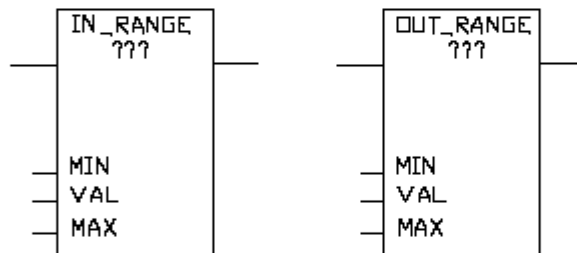


- I0.0 (contatto comandato dalla fotocellula d'ingresso)
- I0.1 (contatto comandato dalla fotocellula di uscita)
- I0.2 (pulsante di reset)
- I0.3 (pulsante di caricamento)

VALORE COMPRESO NEL CAMPO E VALORE FUORI CAMPO

Queste istruzioni verificano se un valore d'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.



Per selezionare il tipo di dati fare clic su “???”

TIPI DI DATI PER I PARAMETRI

PARAMETRO	TIPO DI DATI	DESCRIZIONE
MIN, VAL, MAX	SInt, Int, DInt, UInt, UInt, UDIInt, Real, LReal, costante. (vedi tipi di dati)	Ingressi del comparatore

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

Il confronto IN_RANGE è vero se: $MIN \leq VAL \leq MAX$.

Il confronto OUT_RANGE è vero se: $VAL < MIN$ o $VAL > MAX$.

2.8

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.

TIPI DI DATI BOOL, BYTE, WORD E DWORD

Tipo di dati	Dimensione (bit)	Tipo di numero	Campo numerico
Bool	1	Booleano Binario	Vero o falso 0 o 1
Byte	8	Binario Numero intero senza segno	2#0.....2#11111111 0.....255
Word	16	Binario Numero intero senza segno	2#0.....2#1111111111111111 0.....65535
Dword	32	Binario Numero intero senza segno	2#0.....2#11111111111111111111111111111111 0.....4294967295

Nota: sono stati omessi i numeri in base ottale ed esadecimale.

TIPI DI NUMERI INTERI

Tipo di dati	Dimensione (bit)	Campo numerico
USInt	8	0.....255
SInt	8	Da -128 a 127
UInt	16	Da 0 a 65535
Int	16	Da -32768 a 32767
UDInt	32	Da 0 a 4.294.967.295
DInt	32	Da -2.147.483.648 a 2.147.483.647

TIPI DI DATI REALI IN VIRGOLA MOBILE

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 a 6 cifre significative, mentre i numeri in virgola mobile a precisione doppia sono precisi fino a 15 cifre significative.

Tipo di dati	Dimensioni (bit)	Campo numerico
Real	32	-3,402823e ⁺³⁸-1,175495e ⁻³⁸ , ±0, +1,175495e ⁻³⁸+3,402823e ⁺³⁸
LReal	64	-1,7976931348623158e ⁺³⁰⁸-2,2250738585072014e ⁻³⁰⁸ , ±0, +2,2250738585072014e ⁻³⁰⁸+1,7976931348623158e ⁺³⁰⁸

Nota: la e che compare nel campo numerico significa elevato.

TIPI DI DATI DI ORA E DATA

Tipo di dati	Dimensione	Campo
Time	32 bit	T#-24d_20h_31m_23s_648ms..... T#24d_20h_31m_23s_647ms
Date	32 bit	D#1990-1-1.....D#2168-12-31
Time_of_Day	32 bit	TOD#0:0:0.0.....TOD#23:59:59.999
DTL (Date and Time Long)	12 byte	Min: DTL#1970-01-01-00:00:00.0 Max: DTL#2554-12-31-23:59:59.999999999

TIME

I dati TIME sono memorizzati come numero intero doppio con segno espresso in millisecondi. Il formato dell'editor può usare le informazioni per giorno (d), ore (h), minuti (m), secondi (s) e millisecondi (ms).

Non è necessario specificare tutte le unità di tempo.

DATE

I dati DATE sono memorizzati come numero intero senza segno che viene interpretato come il numero di giorni aggiunti alla data di base 01/01/1990 per ottenere la data specificata.

Il formato dell'editor deve specificare anno, mese, e giorno.

TOD

I dati TOD (TIME_OF_DAY) sono memorizzati come numero intero doppio senza segno che viene interpretato come il numero di millisecondi a partire dalla mezzanotte per l'ora del giorno specificata (mezzanotte = 0 ms).

Devono essere specificati ora (24ore/giorno), minuti e secondi.

La specifica della frazione di secondo è opzionale.

DTL

Il tipo di dati DTL (Date and Time Long) utilizza una struttura a 12 byte che salva le informazioni su data e ora.

I dati DTL possono essere definiti sia nella memoria temporanea di un blocco che in un DB.

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.

ELEMENTI DELLA STRUTTURA DEL DTL

Byte	Componente	Tipo di dati	Campo di valori
0-1	Anno	UInt	1970.....2554
2	Mese	UInt	1.....12
3	Giorno	UInt	1.....31
4	Giorno della settimana*	UInt	1 (domenica)..7 (sabato)
5	Ora	UInt	0.....23
6	Minuto	UInt	0.....59
7	Secondo	UInt	0.....59
8-9-10-11	Nanosecondi	UDInt	0.....999999999

*Nota: il giorno della settimana non viene considerato durante l'immissione dei valori.