

P I X S Y S
elettronica

**OPERATOR PANEL
TOUCH SCREEN**

TD320

**Manuale
User manual**

SOMMARIO

Introduzione	5
Identificazione del modello	5
1 Dimensioni meccaniche ed installazione	6
2 Caratteristiche Display	7
3 Collegamenti elettrici	8
3.1 Collegamenti morsettiera	8
3.2 Porte seriali di comunicazione	9
3.2.1 COM1 su connettore DB25 poli	11
3.2.2 EXP1 su connettore DB9 e DB25 poli.....	12
3.2.3 COM2 interfaccia RS232	14
3.3 Ingressi digitali NPN	15
4 Programmazione del terminale	16
4.1 Starter Kit – Connessione del terminale al PC.....	17
4.2 L'ambiente di sviluppo	18
4.2.1 Creazione di un nuovo progetto	20
4.2.2 Modifica di un progetto già esistente.....	27
5 Aree di memoria del TD320	28
5.1 Area Variabili V	28
5.2 Area Special Marker SM	29
5.3 Area Ingressi Digitali I	44
5.4 Area Uscite Digitali Q.....	44
5.5 Area Marker M	44
5.6 Area Ingressi Analogici AI	45
5.7 Area Uscite Analogiche AQ	45
5.8 Aree Timer T e Preset Timer PT	45
5.9 Area Contatori C e Preset Contatori PV.....	45
5.10 Area Relè Bistabili B	46
5.11 Area EEProm	46
5.12 Area MMC.....	46
5.13 Area TX/RX COM1	47
5.14 Area TX/RX EXP1	47
6 Protocolli di comunicazione	48
6.1 Gestione porte di comunicazione.....	48
6.1.1 Porte COM1 ed EXP1	49
6.1.2 Porta COM2	49

6.2	Protocollo ModBus RTU	49
6.2.1	ModBus RTU Master	50
6.2.2	ModBus RTU Slave	51
6.3	Protocollo NAIS Matsushita Master	55
7	Programmazione Ladder del TD320	59
7.1	Contatti ingressi digitali I	59
7.2	Contatti uscite digitali Q	59
7.3	Relè bistabili B	59
7.4	Timer T	60
7.5	Contatori C	61
7.6	Formule matematiche FM	62
7.7	Assegnazioni MOV	62
7.8	Assegnazioni multiple BLKMOV	62
7.9	Assegnazioni indicizzate MOVIND	63
7.10	Assegnazioni MOVTEXT	63
7.11	Contatti ingressi digitali immediati II	63
7.12	Contatti IF	64
7.13	Funzioni SBIT e RBIT	64
7.14	Contatti BIT	64
7.15	Funzioni RANGE	64
7.16	Contatti NOT	65
7.17	Contatti P e N	65
7.18	Funzioni SEND	65
7.19	Funzioni TUNE POS e POS	66
7.20	Funzioni COM ed EXP	69
7.21	Funzioni StartPID, PID e SetOutPID	70
7.22	Funzioni GENSET	73
7.23	Funzioni CONV	75
8	Note / Aggiornamenti	77

Introduzione

Grazie per aver scelto uno strumento Pixsys.

Il modello TD320 è un terminale grafico touch screen con PLC integrato adatto per la supervisione ed il controllo di impianti dove è necessario l'intervento dell'operatore (HMI).

Le risorse grafiche sono facilmente gestibili da TdDesigner, un ambiente di sviluppo semplice e versatile, mentre la logica relativa al PLC è gestibile dall'ambiente di sviluppo PLProg, comune ad altri dispositivi Pixsys (PL250, TCT500, ecc.).

La protezione del frontale è IP54, del contenitore IP30.

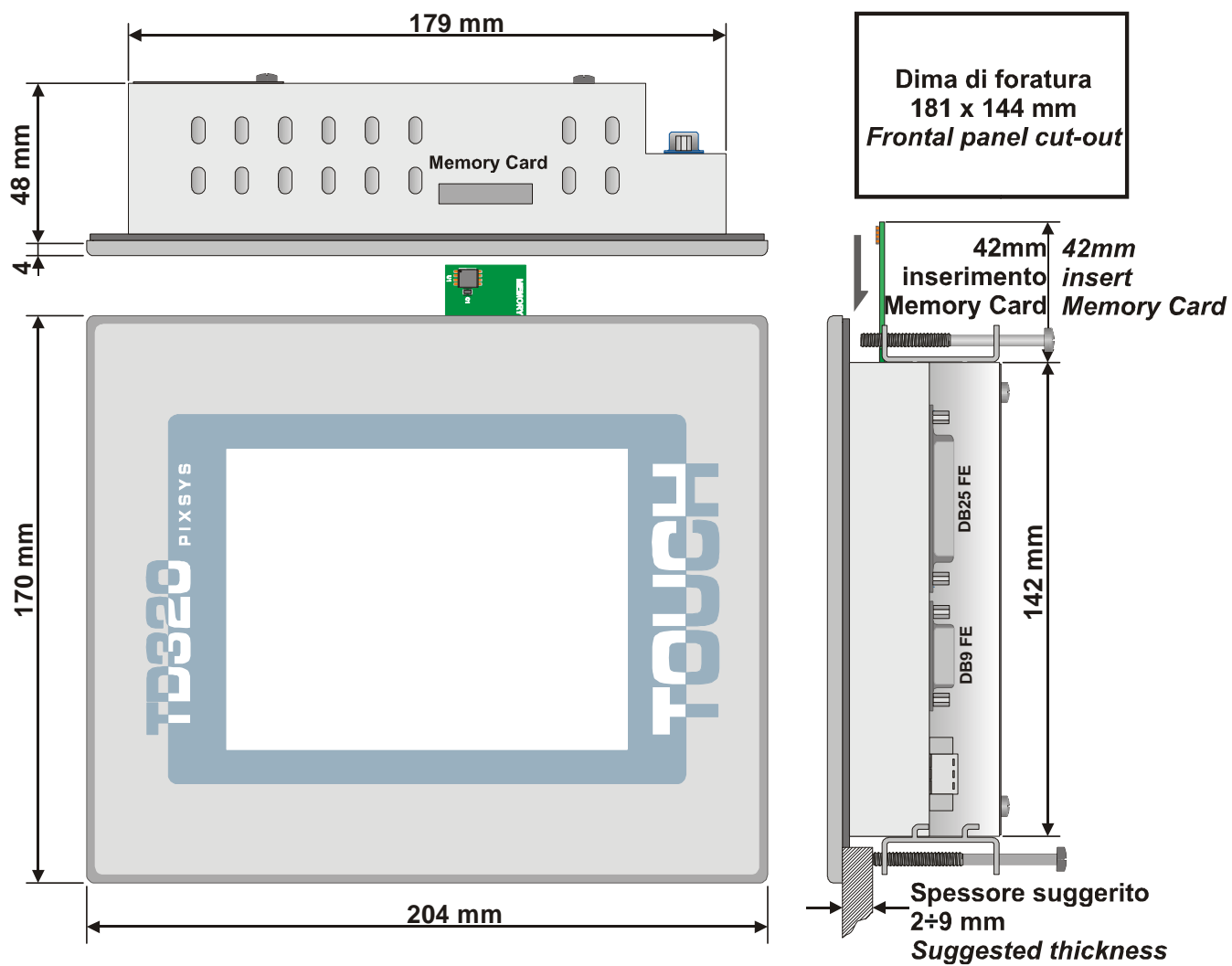
Identificazione del modello

E' prevista una sola versione del terminale TD320, in bassa tensione continua o alternata. Segue la composizione della sigla.

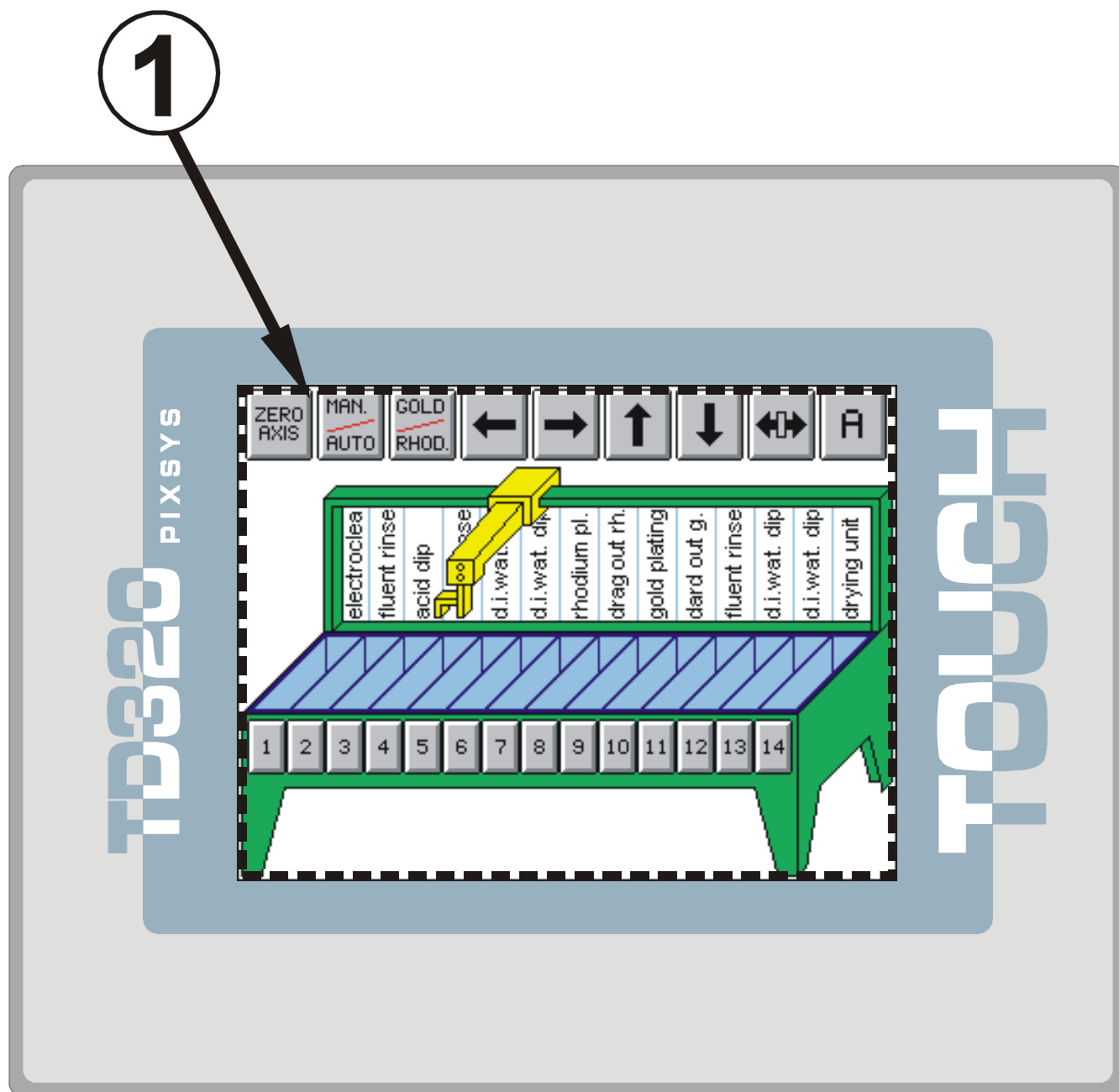
Composizione della sigla

TD320- Alimentazione	<input type="checkbox"/>	12...24V AC/DC ±15% 50/60Hz
	AD	

1 Dimensioni meccaniche ed installazione



2 Caratteristiche Display



1	DISPLAY	Tipo: LCD touch screen resistivo STN retroilluminato Dimensioni: Area attiva 5.7" 115.18(W)mm x 86.38(H)mm Risoluzione: 320x240 pixel Colori: 256 (8bit) Immagini importabili: bitmap a 256 colori(.bmp)
---	----------------	--

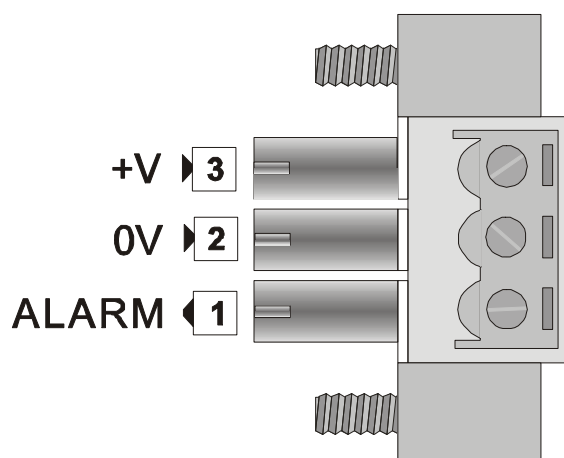
3 Collegamenti elettrici



Benché questo strumento sia stato progettato per resistere ai più gravosi disturbi presenti in ambienti industriali è buona norma seguire la seguenti precauzioni:

- Distinguere la linea di alimentazioni da quelle di potenza.
- Evitare la vicinanza di gruppi di tele ruttori, contattori elettromagnetici, motori di grossa potenza.
- Evitare la vicinanza di gruppi di potenza in particolare se a controllo di fase.

3.1 Collegamenti morsettiera



Alimentazione

SUPPLY
12 to 24V ac/dc

+ 3
- 2

- 12...24V AC/DC $\pm 15\%$ 50/60Hz

Uscita Allarme

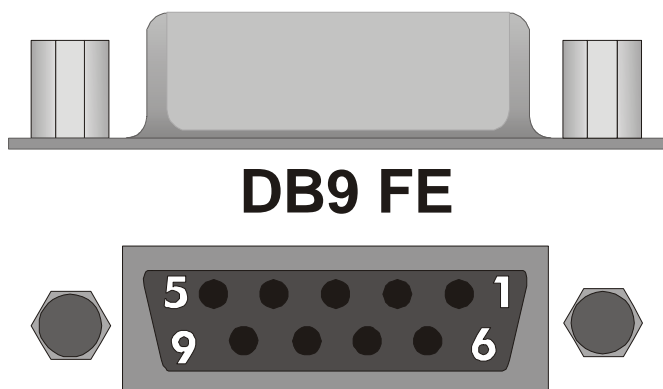


Con contatto attivo (portata contatto **3A/250V~** carico resistivo) è disponibile la tensione **+V** (aliment.) tra i morsetti 1(+) e 2(-)

3.2 Porte seriali di comunicazione

La comunicazione del terminale TD320 con altri dispositivi è possibile tramite connessione seriale su interfacce **RS485**, **RS232** ed **RS422**.

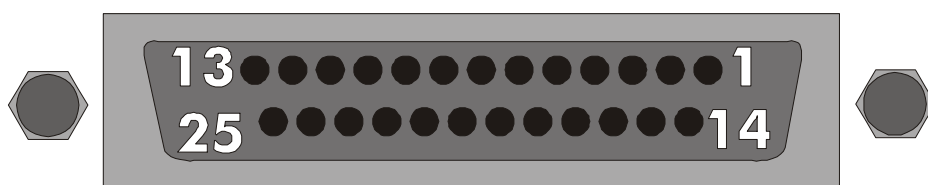
I segnali elettrici sono disponibili in due connettori presenti nel retro del terminale: **DB9 poli** e **DB25 poli**



CONNETTORE	NUM. PIN	SEGNALE	PORTA
DB9 POLI	1	Non utilizzato	-
	2	RX – RS232 Program	COM2
	3	TX – RS232 Program	COM2
	4	RS485 -	EXP1
	5	GND RS485 / RS232	EXP1 / COM2
	6	TX – RS232	EXP1
	7	RX – RS232	EXP1
	8	Non utilizzato	-
	9	RS485 +	EXP1



DB25FE

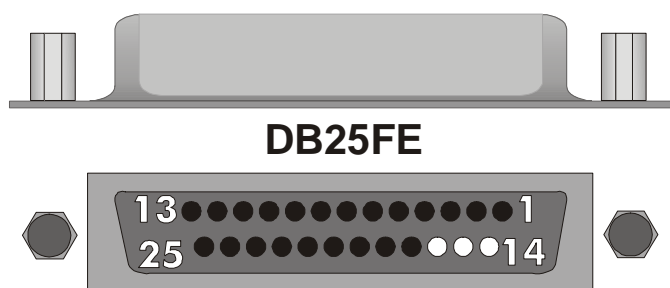


CONNETTORE	NUM. PIN	SEGNALE	PORTA
DB25 POLI	1	Non utilizzato	-
	2	Non utilizzato	-
	3	Non utilizzato	-
	4	Non utilizzato	-
	5	GND (Com. ingressi digitali)	-
	6	DI1 (ingresso digitale NPN)	-
	7	GND – RS232	EXP1 / COM2
	8	DI2 (ingresso digitale NPN)	-
	9	DI3 (ingresso digitale NPN)	-
	10	DI4 (ingresso digitale NPN)	-
	11	RX – RS232 Program	COM2
	12	TX – RS232 Program	COM2
	13	DI5 (ingresso digitale NPN)	-
	14	GND isolata RS485 / RS422	COM1
	15	RS485+ / TX+ RS422	COM1
	16	RS485- / TX- RS422	COM1
	17	RX+ RS422	COM1
	18	RX- RS422	COM1
	19	DI6 (ingresso digitale NPN)	-
	20	DI7 (ingresso digitale NPN)	-
	21	RX – RS232	EXP1
	22	TX – RS232	EXP1
	23	DI8 (ingresso digitale NPN)	-
	24	Non utilizzato	-
	25	Non utilizzato	-

3.2.1 COM1 su connettore DB25 poli

La porta di comunicazione **COM1** è disponibile nel connettore a 25 poli nell'interfaccia RS485 oppure RS422 (protocollo, baud rate e formato impostabili).

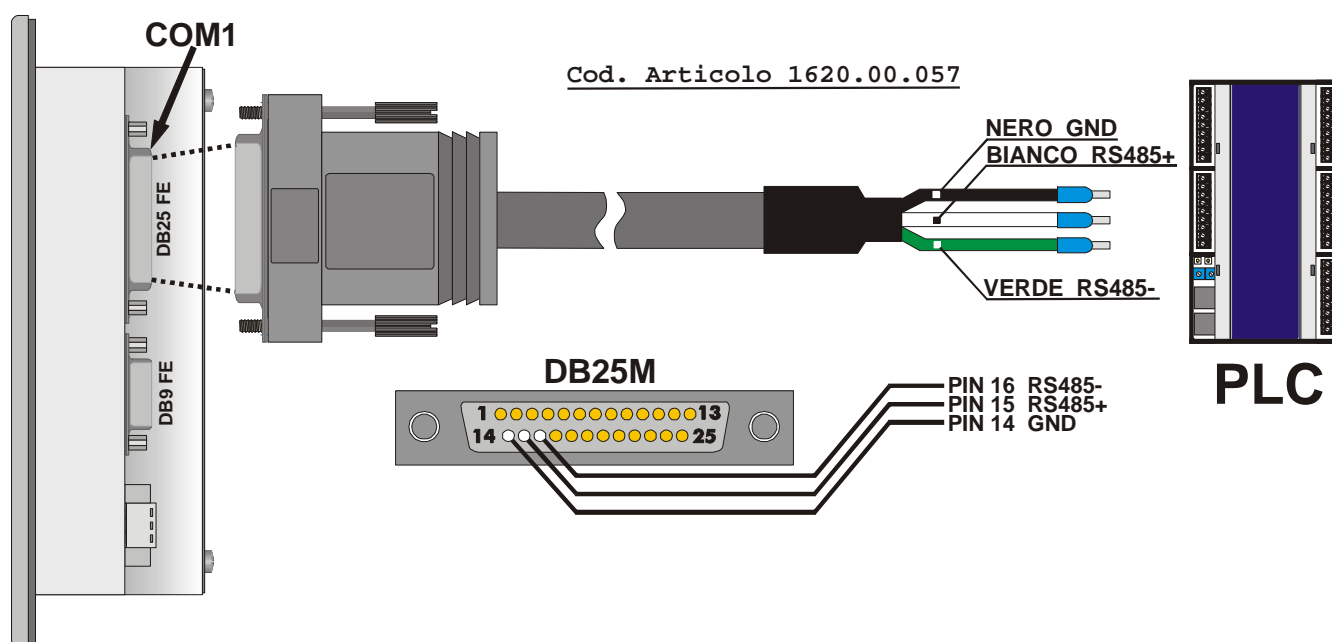
3.2.1.1 Interfaccia RS485



Interfaccia RS485 su DB25 (COM1)		
○	PIN 14	GND
○	PIN 15	RS485+
○	PIN 16	RS485-

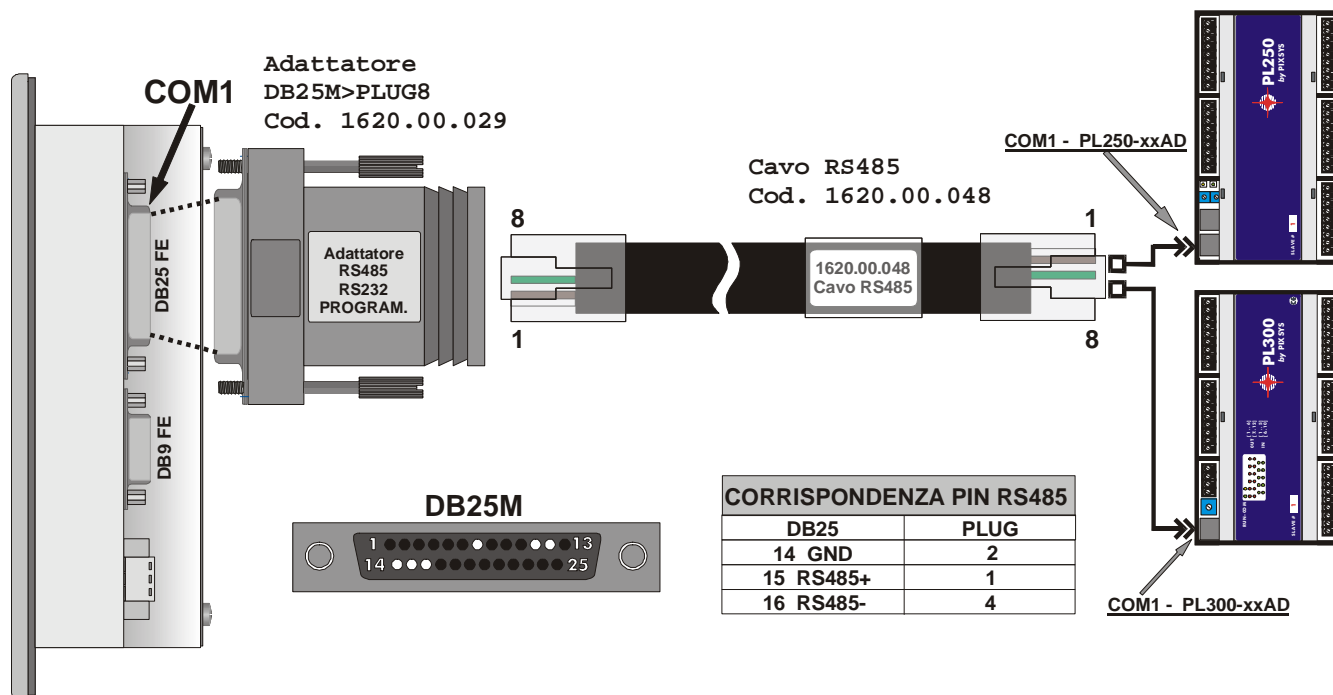
3.2.1.1.1 Cavo di com. COM1 in RS485 generico

E' a disposizione un cavo (**cod. art. 1620.00.057**, opzionale), che dal connettore DB25 porta fuori la **COM1** in RS485 per una generica connessione con altri dispositivi (per i particolari riguardanti i protocolli di comunicazione, consultare altra documentazione).

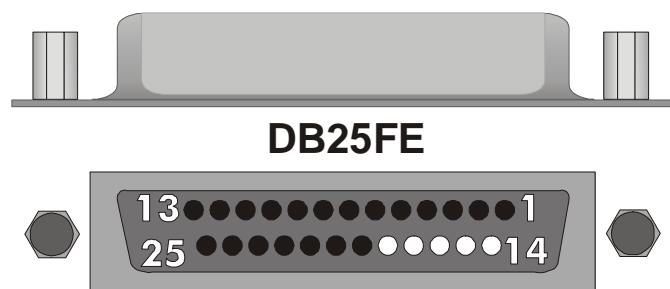


3.2.1.1.2 Cavo di com. COM1 in RS485 per PL250 / PL300

Per la comunicazione con altri dispositivi Pixsys (**PL250-XXAD** e **PL300-XXAD**) è a disposizione (opzionale) un cavo che connette la porta **COM1** in RS485 dal connettore DB25 del terminale alla porta COM1 su PLUG dei PLC.



3.2.1.2 Interfaccia RS422

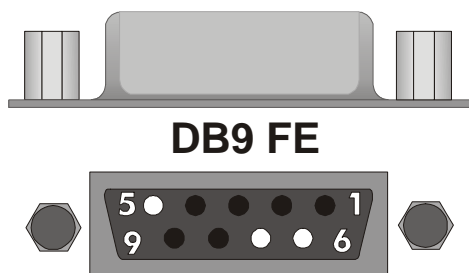


Interfaccia RS422 su DB25 (COM1)		
<input type="radio"/>	PIN 14	GND
<input type="radio"/>	PIN 15	TX+
<input type="radio"/>	PIN 16	TX-
<input type="radio"/>	PIN 17	RX+
<input type="radio"/>	PIN 18	RX-

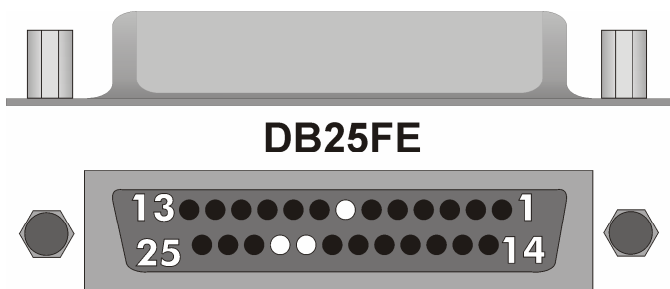
3.2.2 EXP1 su connettore DB9 e DB25 poli

La porta di comunicazione **EXP1** è disponibile nel connettore a 9 poli nell'interfaccia RS232 o RS485 e nel connettore a 25 poli nell'interfaccia RS232 (protocollo, baud rate e formato impostabili)

3.2.2.1 Interfaccia RS232

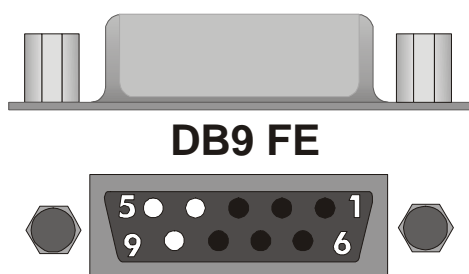


Interfaccia RS232 su DB09 (EXP1)		
<input type="radio"/>	PIN 5	GND
<input type="radio"/>	PIN 6	TX - RS232
<input type="radio"/>	PIN 7	RX - RS232



Interfaccia RS232 su DB25 (EXP1)		
<input type="radio"/>	PIN 7	GND
<input type="radio"/>	PIN 22	TX - RS232
<input type="radio"/>	PIN 21	RX - RS232

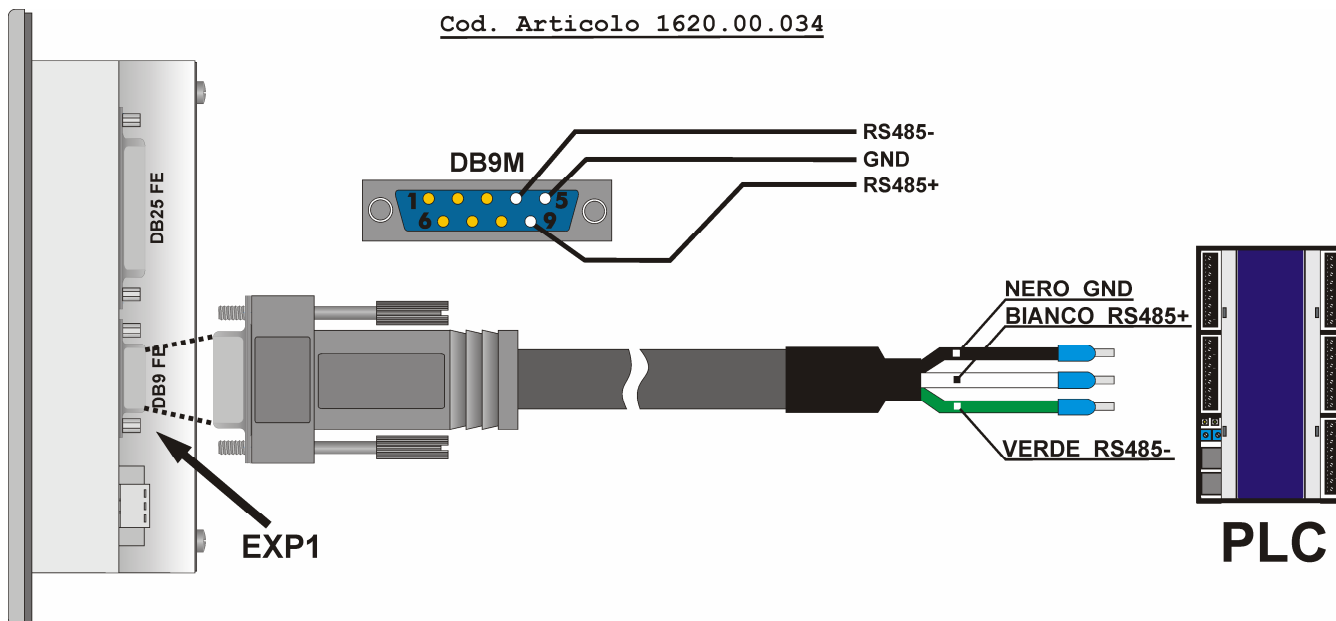
3.2.2.2 Interfaccia RS485



Interfaccia RS485 su DB09 (EXP1)		
<input type="radio"/>	PIN 5	GND
<input type="radio"/>	PIN 9	RS485+
<input type="radio"/>	PIN 4	RS485-

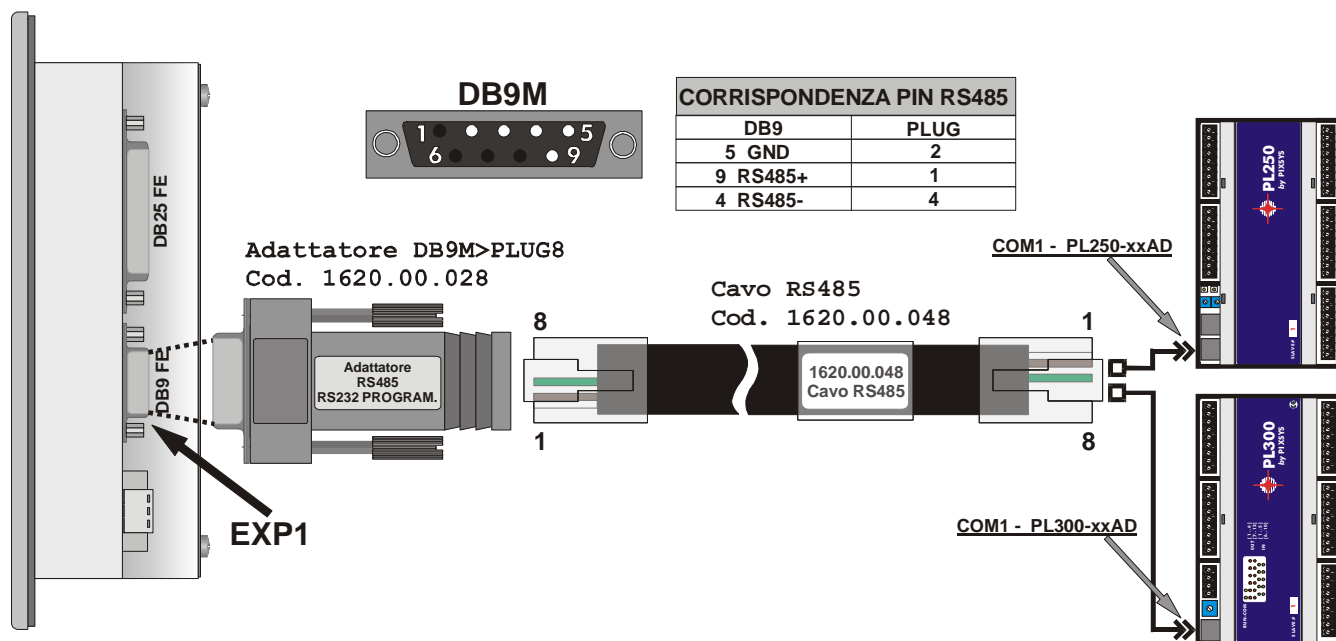
3.2.2.2.1 Cavo di com. EXP1 in RS485 generico

E' a disposizione un cavo (**cod. art. 1620.00.034**, opzionale), che dal connettore DB9 porta fuori la **EXP1** in RS485 per una generica connessione con altri dispositivi (per i particolari riguardanti i protocolli di comunicazione, consultare altra documentazione).



3.2.2.2 Cavo di com. EXP1 in RS485 per PL250 / PL300

Per la comunicazione con altri dispositivi Pixsys (**PL250-XXAD** e **PL300-XXAD**) è a disposizione (opzionale) un cavo che connette la porta **EXP1** in RS485 dal connettore DB9 del terminale alla porta COM1 su PLUG dei PLC.



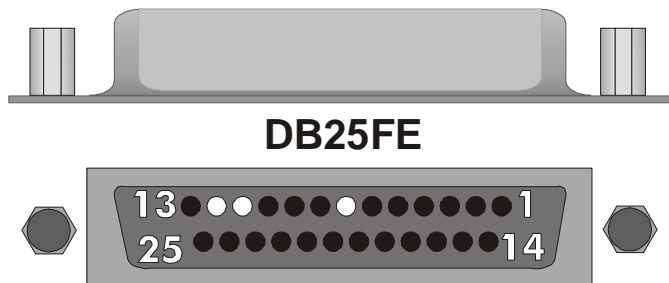
3.2.3 COM2 interfaccia RS232

La porta di comunicazione **COM2** è disponibile sia nel connettore a 25 poli che nel connettore a 9 poli, ma solo nell'interfaccia

RS232 (protocollo **MODBUS SLAVE**, formato **8,N,1**, baud rate impostabile).

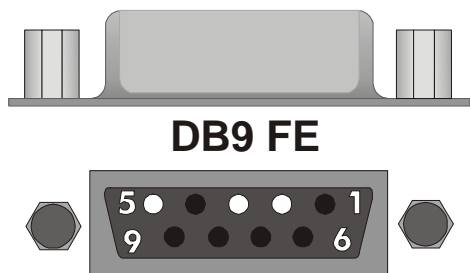
Solitamente è la porta di comunicazione utilizzata per la programmazione del terminale tramite PC (vedi cap. 4).

3.2.3.1 COM2 su connettore DB25 poli



Interfaccia RS232 su DB25 (COM2)		
<input type="radio"/>	PIN 7	GND
<input type="radio"/>	PIN 12	TX - RS232
<input type="radio"/>	PIN 11	RX - RS232

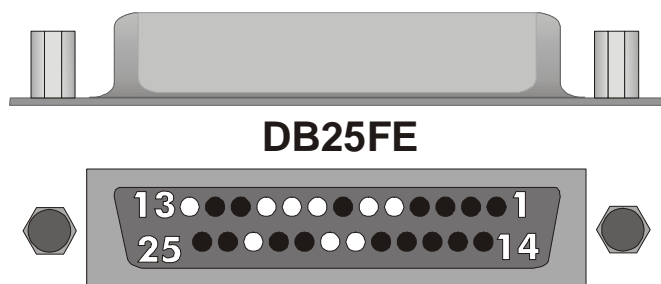
3.2.3.2 COM2 su connettore DB9 poli



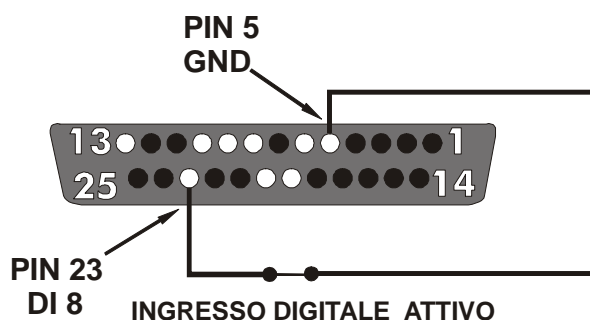
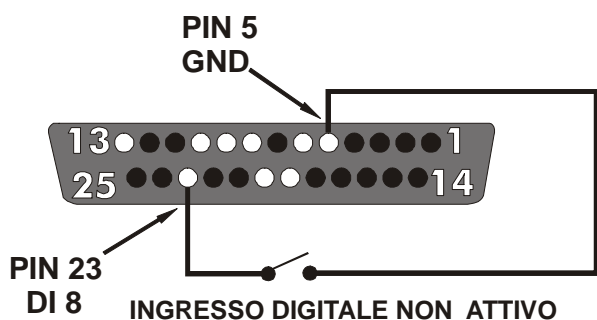
Interfaccia RS232 su DB09 (COM2)		
<input type="radio"/>	PIN 5	GND
<input type="radio"/>	PIN 3	TX - RS232
<input type="radio"/>	PIN 2	RX - RS232

3.3 Ingressi digitali NPN

Nel connettore DB25 poli sono presenti 8 ingressi digitali NPN. L'ingresso digitale è attivo se il rispettivo PIN è cortocircuitato con GND (PIN 5, riferimento).



Ingressi digitali NPN		
<input type="radio"/>	PIN 5	Riferimento
<input type="radio"/>	PIN 6	DI 1
<input type="radio"/>	PIN 8	DI 2
<input type="radio"/>	PIN 9	DI 3
<input type="radio"/>	PIN 10	DI 4
<input type="radio"/>	PIN 13	DI 5
<input type="radio"/>	PIN 19	DI 6
<input type="radio"/>	PIN 20	DI 7
<input type="radio"/>	PIN 23	DI 8

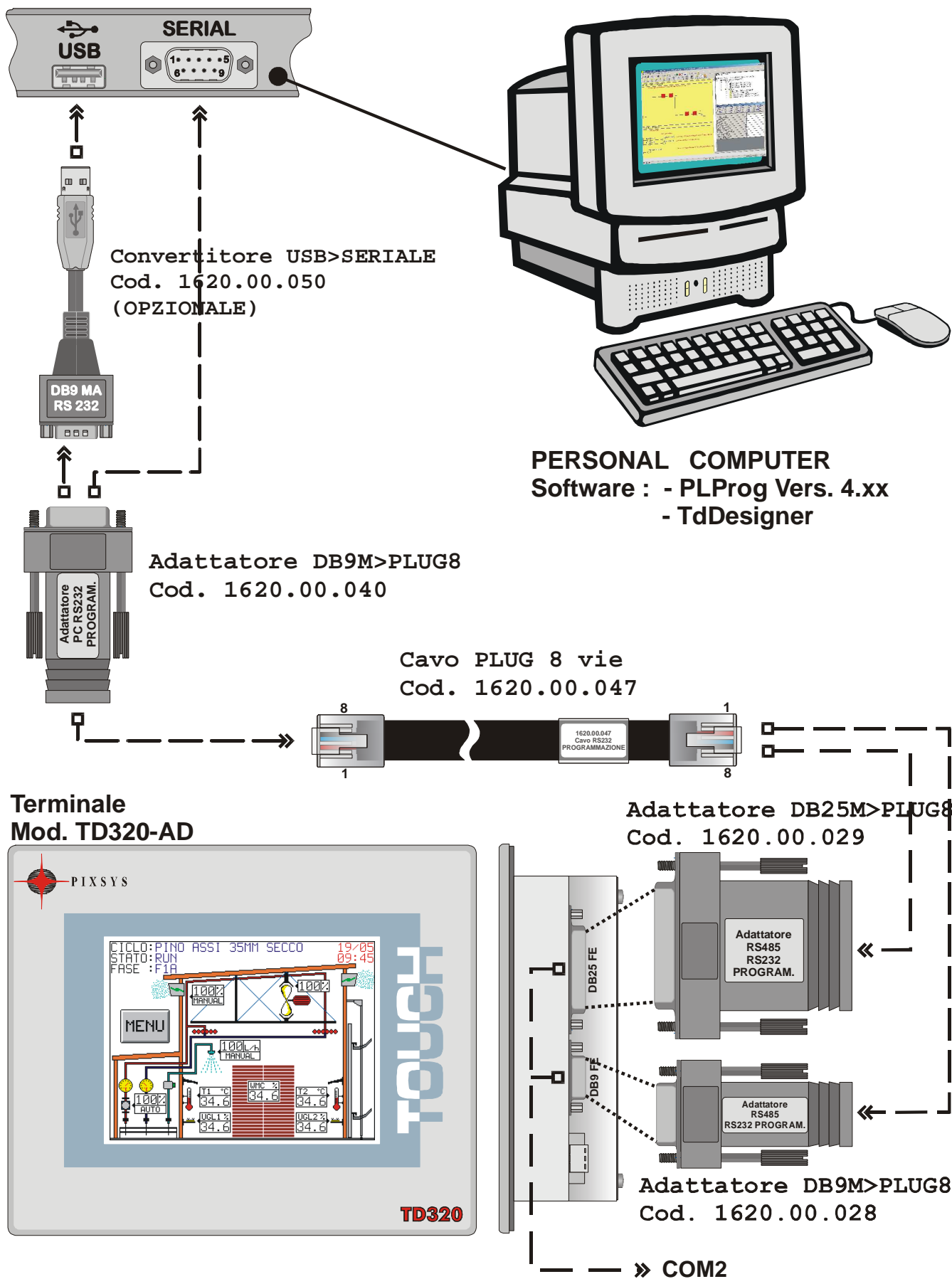


4 Programmazione del terminale

Per programmare il terminale è necessario connetterlo ad un PC. Il kit di sviluppo (opzionale, **cod.art. 2100.10.008**) mette a disposizione il cavo necessario per la connessione e l'ambiente di sviluppo per creare le applicazioni.

La programmazione occupa la porta di comunicazione **COM2**, presente in entrambi i connettori, ma solo nell'interfaccia RS232. Gli adattatori dal lato del terminale sono 2 per permettere all'utente di programmare il terminale tramite il connettore DB25 oppure tramite il DB9.

4.1 Starter Kit – Connessione del terminale al PC



4.2 L'ambiente di sviluppo

Il TD320 è un terminale grafico HMI con PLC integrato. Consente di centralizzare tutta la logica di funzionamento del sistema che si deve supervisionare e controllare.

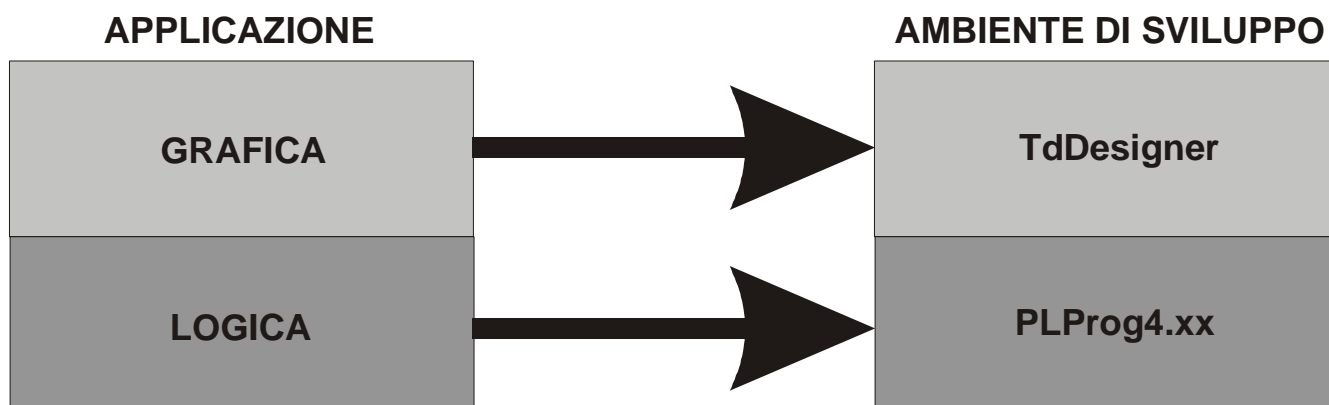
La parte grafica dell'ambiente di sviluppo deve gestire le pagine visualizzabili ed i loro *item* fondamentali (sinottici, pulsanti, edit numerici e di testo, immagini) e l'interazione tra i vari *oggetti* e le aree di memoria (le aree di memoria a cui devono fare riferimento pulsanti, testi ed immagini).

La logica di funzionamento dell'impianto, cioè il modo in cui le aree di memoria devono interagire tra loro, è invece gestita dal PLC.

Il terminale TD320 è anche un PLC, quindi da un lato si gestirà la grafica, dall'altro si potrà gestire la logica, dando la possibilità agli altri PLC connessi il solo compito di “reperire le informazioni” (ingressi digitali ed analogici, encoder ecc.) e di “attivare gli attuatori” (uscite digitali, analogiche, ecc.).

L'ambiente di sviluppo è suddiviso in due sotto-ambienti:

- **TdDesigner:** gestisce tutte le risorse strettamente legate alla grafica.
- **PLProg:** gestisce le interazioni tra le aree di memoria del terminale (codice Ladder, comune ad altri PLC Pixsys, es. PL250 e TCT500).



Qualsiasi applicazione gestita dal terminale TD320 dovrà quindi essere realizzata utilizzando entrambi gli ambienti di sviluppo, **implementando quindi due diversi file** tra loro strettamente collegati.

Il funzionamento del terminale prevede una ripartizione del tempo dedicato alla gestione della grafica (implementata con TdDesigner) e del tempo dedicato alla gestione del PLC (implementata con PLProg 4.xx).

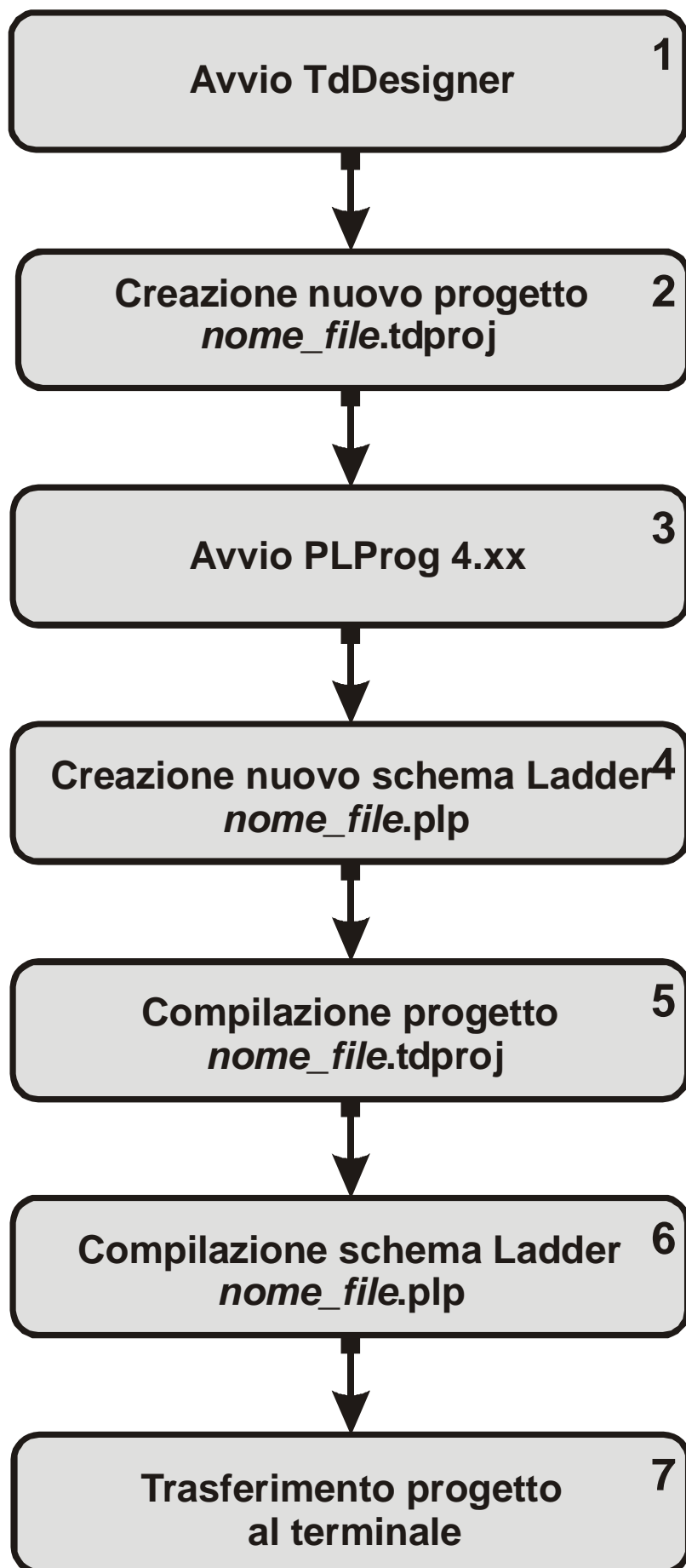
L'impostazione di default prevede una ripartizione equa del ciclo di esecuzione: il terminale per metà tempo eseguirà le istruzioni inerenti la grafica e per metà tempo le istruzioni Ladder del PLC (ciclicamente).



La ripartizione è impostabile da utente (vedi cap. 5). Si riporta di seguito un esempio in cui l'80% del tempo è dedicato alla grafica ed il 20% al PLC.



4.2.1 Creazione di un nuovo progetto

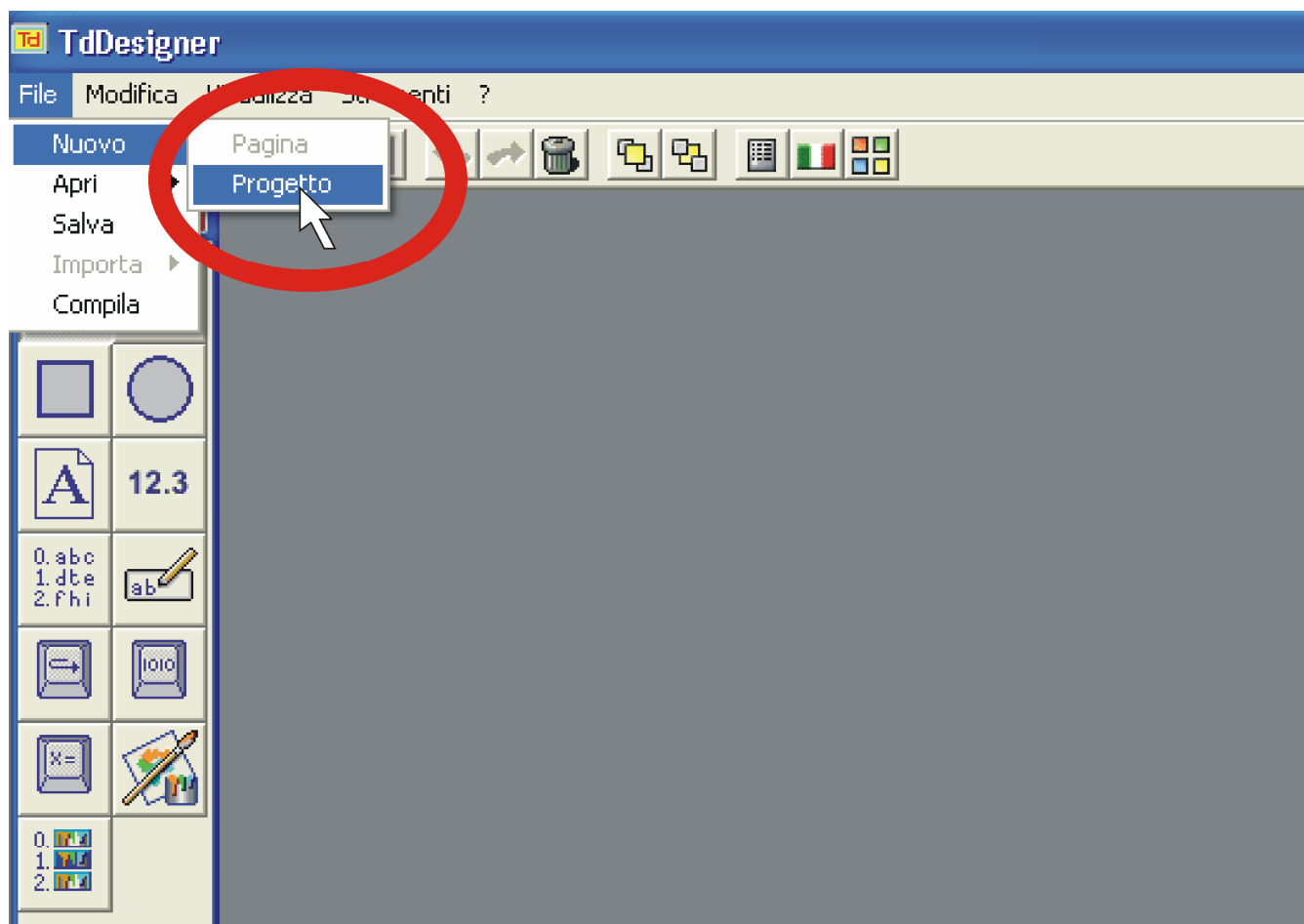


Per realizzare un nuovo progetto e trasferirlo al terminale deve essere seguita la procedura illustrata e descritta di seguito:

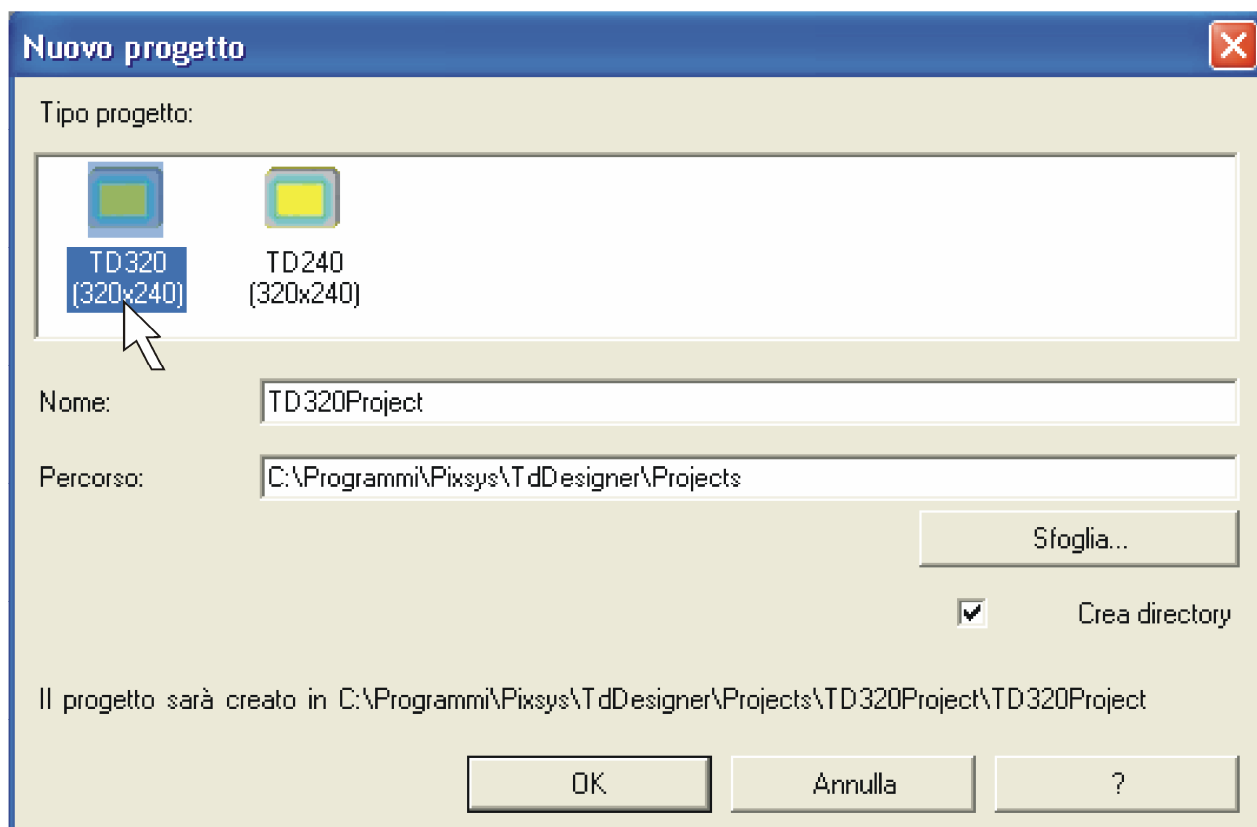
1. **Avvio TdDesigner**: Avviare il software **TdDesigner** dal menu Start\Programmi o dal collegamento sul Desktop (automaticamente creato all'installazione).



2. **Creazione nuovo progetto nome file.tdproj**: Una volta aperto l'ambiente di sviluppo, creare un nuovo progetto come riportato nella figura sottostante:



Selezionare poi il terminale TD320 (320x240 pixel display 5,7")



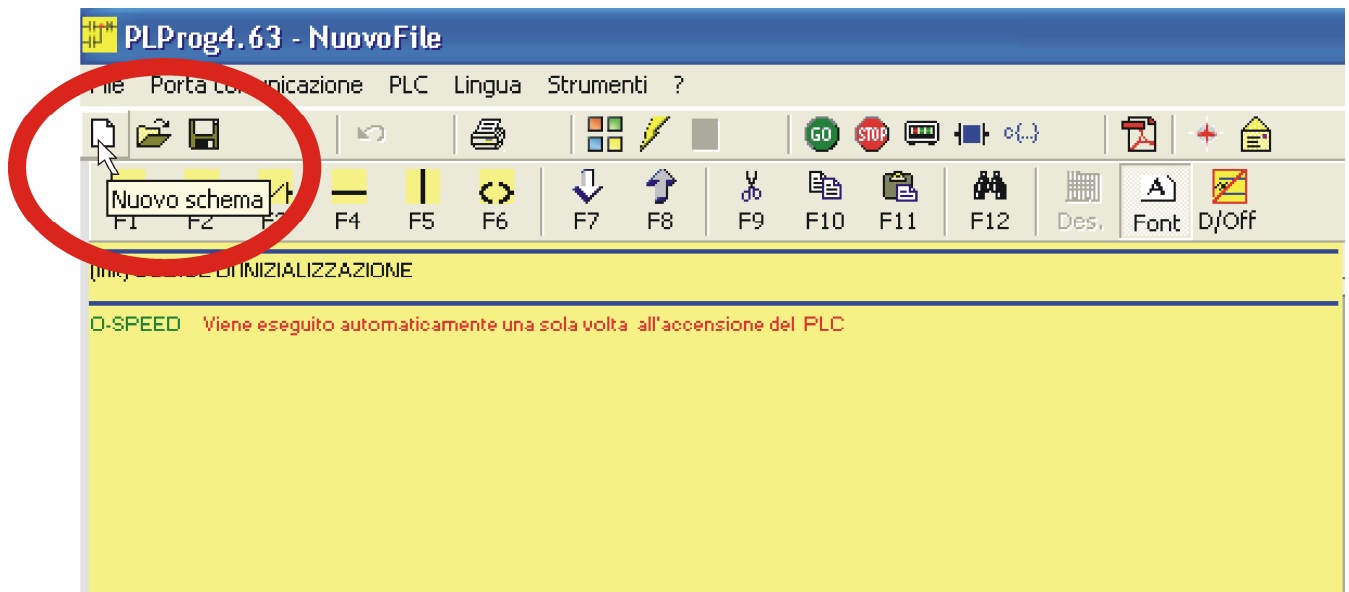
L'ambiente di sviluppo TdDesigner può inserire il nuovo progetto in una directory creata automaticamente oppure in un percorso scelto dall'utente.

La gestione della grafica è affrontata in altra documentazione, disponibile con il kit di sviluppo (**cod. art. 2100.10.008**) e data per acquisita da parte dell'utente.

3. **Avvio PLProg 4.xx**: Avviare il software **PLProg 4.xx** dal menu Start\Programmi o dal collegamento sul Desktop (automaticamente creato all'installazione).



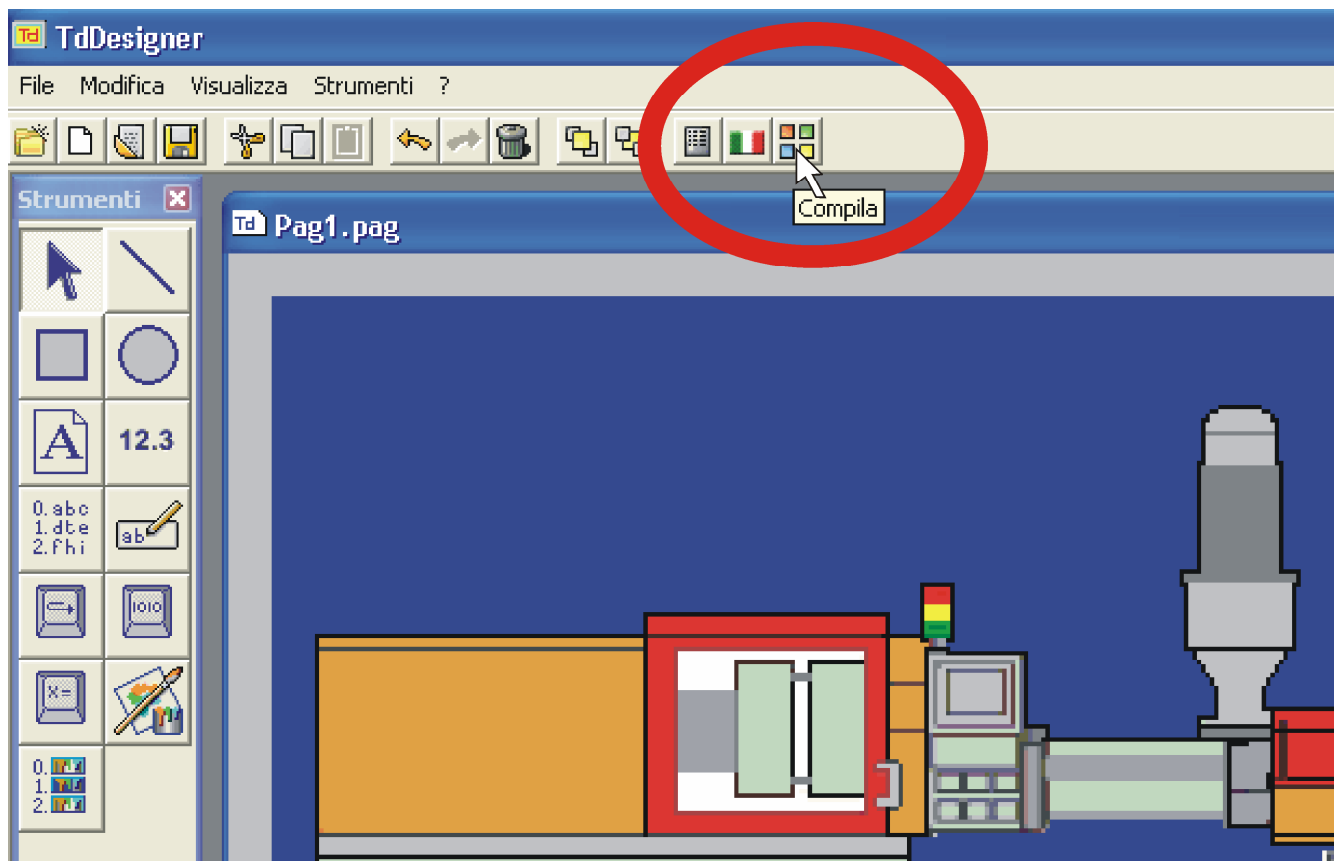
4. **Creazione nuovo schema Ladder *nome_file.plp***: Una volta aperto l'ambiente di sviluppo, creare un nuovo schema come riportato nella figura sottostante:



Apparirà ora una finestra al centro dello schermo: selezionare il terminale TD320 nella voce Seleziona CPU.

La guida al software ed all'implementazione del codice Ladder è disponibile con il kit di sviluppo (**cod. art. 2100.10.008**) e data per acquisita da parte dell'utente.

5. **Compilazione progetto *nome_file.tdproj***: Terminata l'implementazione della grafica, è necessario **compilare** il progetto, come riportato nella figura sottostante (oppure selezionare la voce da File\Compila).

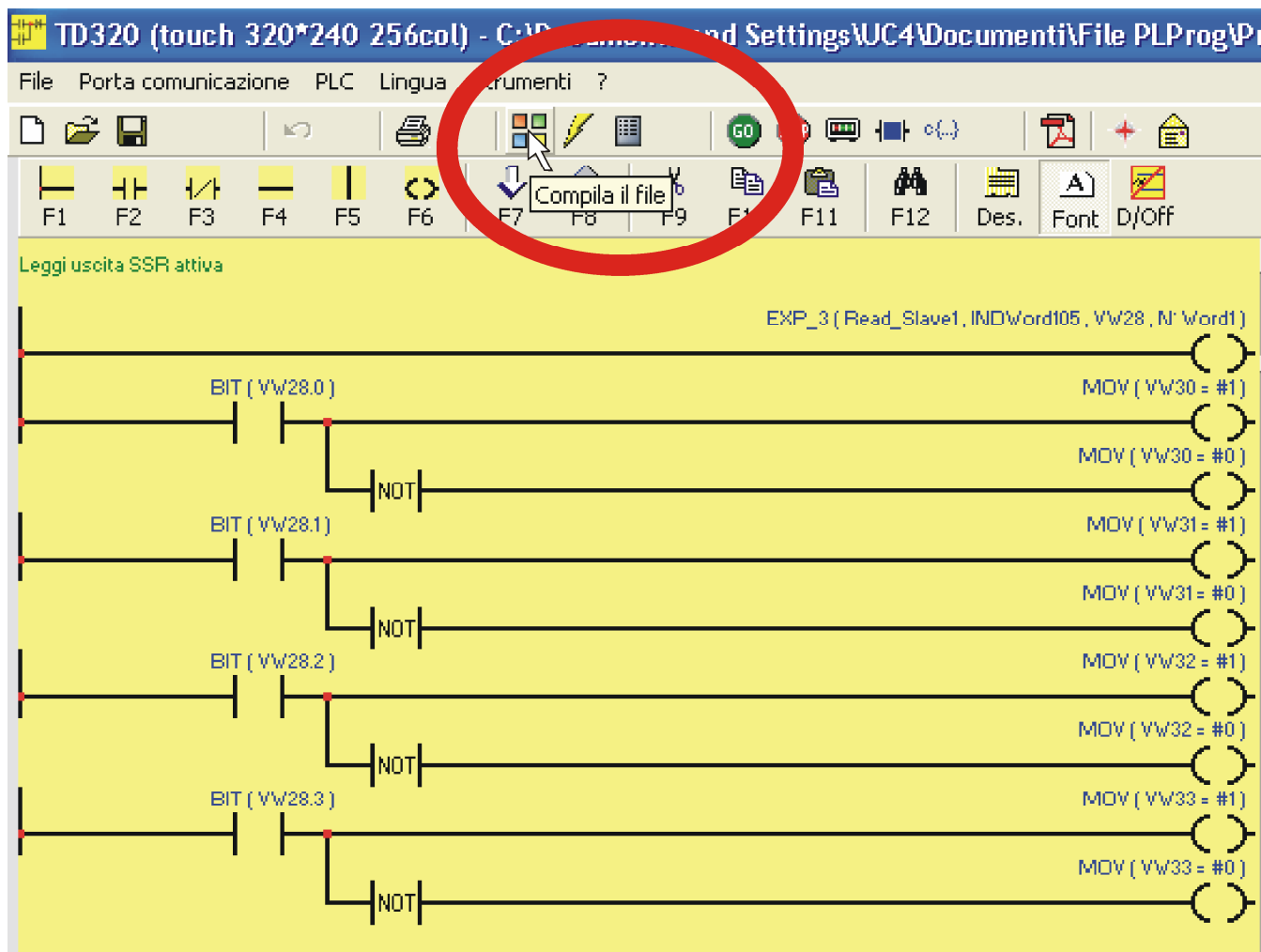


Questa operazione è necessaria per rendere disponibile il progetto appena implementato all'ambiente di sviluppo **PLProg 4.xx**.

La compilazione ha effetto solo se PLProg è aperto ed è stato selezionato il terminale corrispondente come CPU.

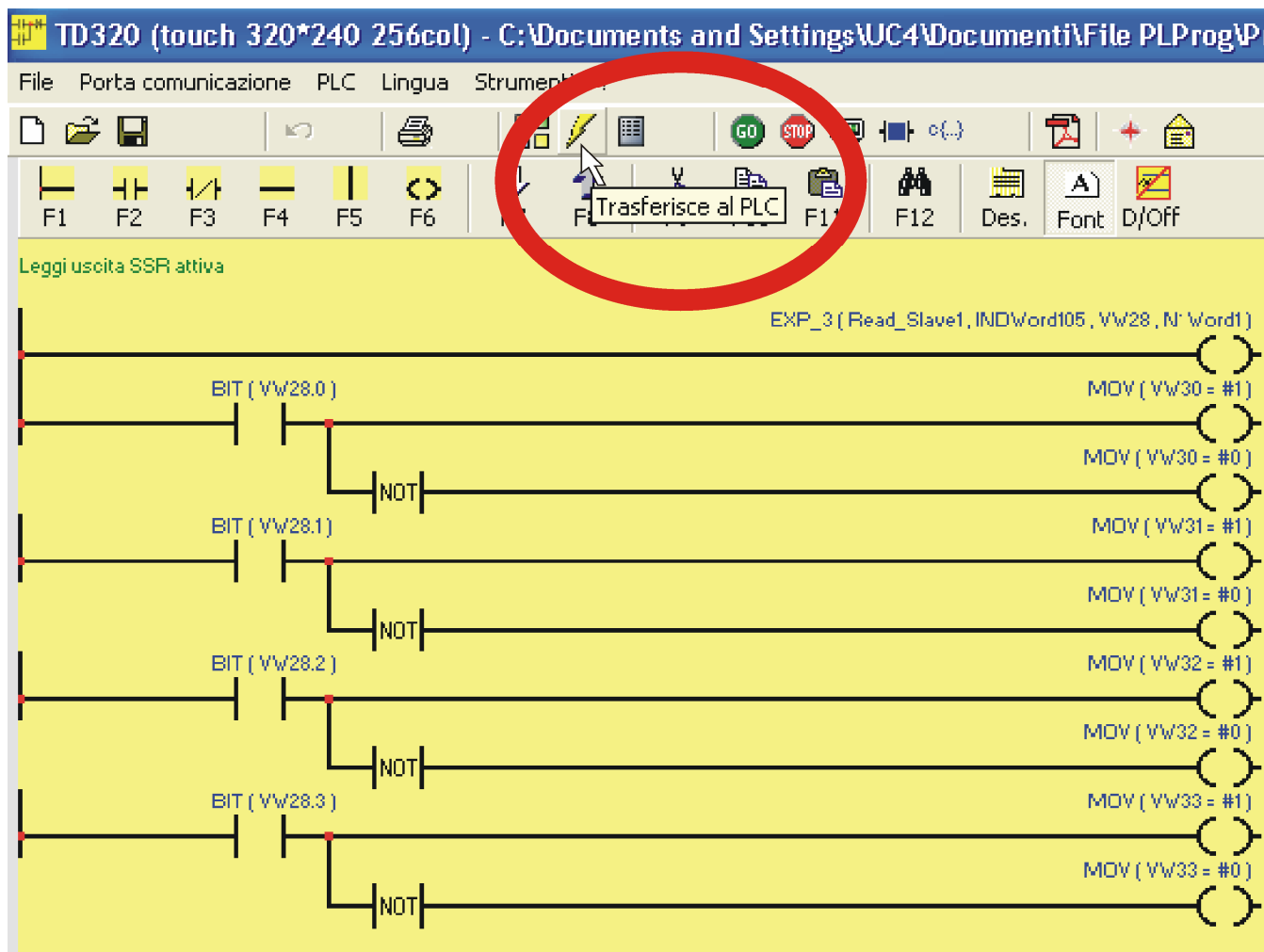
6. **Compilazione schema Ladder *nome file.plp***: Una volta terminata la stesura del diagramma Ladder, è necessario **compilare** il progetto, come riportato nella figura sottostante. E' in questo fondamentale passaggio che si crea il collegamento tra il file Ladder appena compilato nell'ambiente di sviluppo **PLProg4.xx** e quello precedentemente compilato nell'ambiente di sviluppo **TdDesigner**.

Solo con questa operazione sarà infatti possibile comunicare al terminale anche le istruzioni inerenti alla grafica del progetto creato.

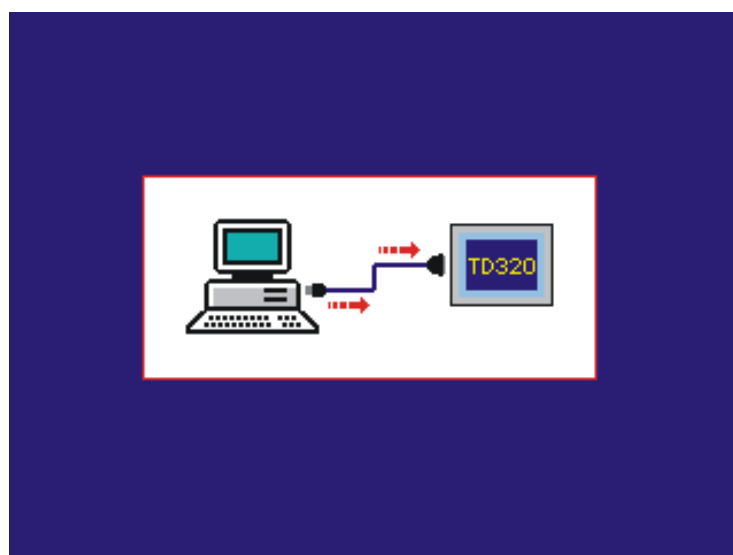


A questo punto, se si salva da PLProg, il file *nome_file.plp* conterrà sia la parte PLC che la parte grafica (non è necessario che il file .tdproj abbia lo stesso nome del file .plp).

7. **Trasferimento progetto al terminale:** Se la compilazione ha avuto esito positivo, si può ora effettuare il download del progetto, come riportato nella figura sottostante. La procedura trasferisce nel terminale sia la parte grafica che la parte PLC.



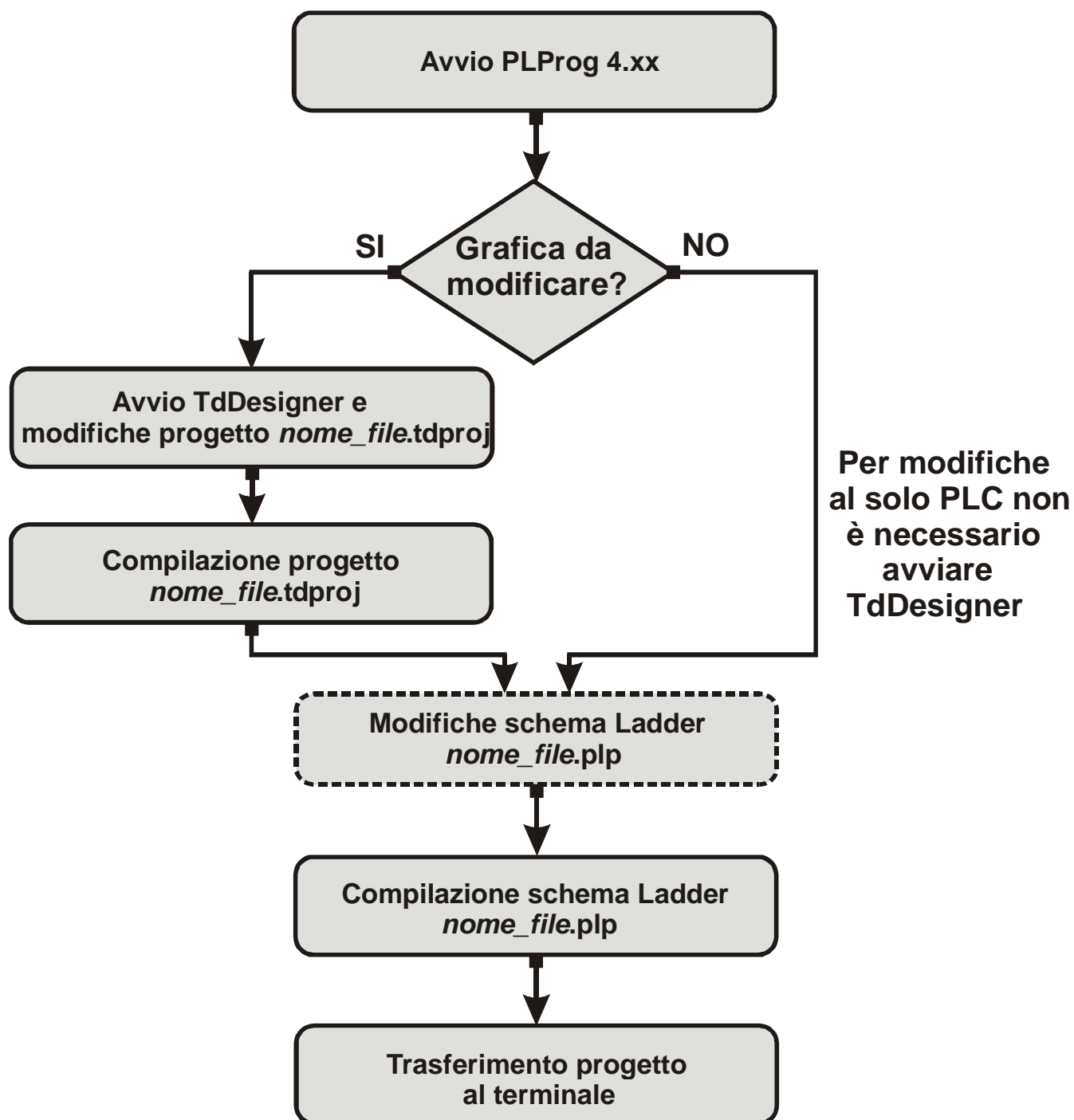
Se il TD320 è correttamente connesso al PC (vedi schema del **par. 4.1**), durante il trasferimento il terminale mostrerà sul display la figura:



Al termine del download, il terminale eseguirà le istruzioni dell'intera applicazione.

4.2.2 Modifica di un progetto già esistente

Nel caso in cui si debba modificare un progetto già esistente, seguire la seguente procedura:



Per le eventuali modifiche della sola parte PLC (tratteggiata) non è necessario avviare TdDesigner. La compilazione del progetto *nome_file.plp* manterrà la grafica invariata e renderà attive le modifiche dello schema Ladder.

5 Aree di memoria del TD320

Il TD320 mette a disposizione aree di memoria dove poter leggere o scrivere i dati del programma. L'accesso alle varie aree può avvenire tramite istruzioni che accedono ai singoli bit (b), a byte (B), a word (W) oppure doppia word (D).

SIGLA	AREA	ACCESSO
V	Area Variabili V	b, W, D
SM	Area Special Marker	b, W, D
I	Area Ingressi Digitali	b, W
AI	Area Ingressi Analogici	b, W
Q	Area Uscite Digitali	b, W
M	Area Marker	b, W
B	Area Bistabili	b
AQ	Area Uscite Analogiche	b, W
T	Area Timer	b, W
PT	Area Preset Timer	b, W
C	Area Contatori	b, W
PV	Area Preset Contatori	b, W
EEP	Area EEPROM	W
MMC	Area EEPROM dati	W
COM1	Area buffer TX/RX porta COM1	B
EXP	Area buffer TX/RX porta EXP1	B

5.1 Area Variabili V

L'area variabili V è una memoria dati a ritenzione utilizzata dal programma per memorizzare i dati delle operazioni.

E' costituita da 10000 locazioni di tipo word (5000 doppie word). L'accesso può avvenire tramite operazioni su bit, word o doppie word. Nell'ultimo caso il numero della doppia word fa sempre riferimento all'organizzazione in word, quindi per accedere a variabili in doppia word consecutive bisogna far avanzare il numero di 2.

I valori memorizzati vengono mantenuti anche in assenza di tensione grazie alla batteria tampone ricaricabile. Una volta carica, la batteria consente di mantenere i dati memorizzati per circa 6 mesi.

ACCESSO A WORD

VW0
VW1
VW2
VW3
VW4
VW5

ACCESSO A DOPPIA WORD

VD0
VD2
VD4

VD1
VD3

5.2 Area Special Marker SM

L'area special marker SM è la memoria a ritenzione dove risiedono tutti i dati necessari al programma ladder per interagire con l'hardware del TD320.

Alcuni vengono inizializzati all'accensione con dei valori di default indicati nella tabella sottostante. In quest'area si trovano le word che gestiscono gli eventi relativi alla grafica, i bit di controllo del PLC e le impostazioni per le porte seriali di comunicazione.

La tabella sottostante descrive il contenuto di ogni singola locazione dell'area special marker, indicando l'indirizzo per l'accesso tramite protocollo ModBus e l'operazione consentita sulla locazione (R = lettura, W = scrittura, R/W = lettura/scrittura).

I bit e le word che non compaiono nelle tabelle non sono utilizzate.

SM N°	ModBus Address	Descrizione / Significato		
SM0	1000	Bit di stato		
		Bit 0	Bit RUN/STOP (1 = RUN). All'accensione questo bit viene sempre forzato ON (PLC in RUN). In STOP le uscite relè del PLC vengono disabilitate.	R/W
		Bit 1	Bit sempre ON per il primo ciclo di scansione del programma principale. Viene utilizzato, ad esempio, per eseguire un sottoprogramma di inizializzazione.	R
		Bit 2	Bit che mette a disposizione un impulso di clock di 60 secondi (ON per 30 secondi, OFF per altri 30).	R
		Bit 3	Bit che mette a disposizione un impulso di clock di 1 secondo (ON per 0,5 secondi, OFF per altri 0,5 secondi)	R
		Bit 4	Bit clock di ciclo di scansione, che è attivo (ON) per un ciclo e disattivo (OFF) per il ciclo successivo. Può essere utilizzato come ingresso di conteggio del ciclo di scansione.	R
		Bit 6	Bit ON durante la fase di trasmissione di dati sulla porta seriale COM1. Viene automaticamente portato a OFF alla fine della trasmissione.	R
		Bit 7	Bit ON durante la fase di trasmissione di dati sulla porta seriale EXP1. Viene automaticamente portato a OFF alla fine della trasmissione.	R
		Bit 8	Bit ON durante la fase di	R

		trasmissione di dati sulla porta seriale COM2. Viene automaticamente portato a OFF alla fine della trasmissione.	
	Bit 9	Questo bit, se impostato ON, abilita per la porta seriale COM1 la modalità "modem". Ciò significa che il timeout tra un carattere e l'altro in ricezione è fissato automaticamente a 40 mS.	R/W
	Bit 10	Questo bit, se impostato ON, abilita per la porta seriale EXP1 la modalità "modem". Ciò significa che il timeout tra un carattere e l'altro in ricezione è fissato automaticamente a 40 mS.	R/W
	Bit 11	Questo bit, se impostato ON, abilita per la porta seriale COM2 la modalità "modem". Ciò significa che il timeout tra un carattere e l'altro in ricezione è fissato automaticamente a 40 mS.	R/W
SM1	1001	Bit diagnostica anomalie / guasti	
	Bit 0	Bit ON in caso di perdita dei dati a ritenzione dell'area special marker.	R/W
	Bit 1	Bit ON in caso di perdita dei dati a ritenzione dell'area variabili V.	R/W
	Bit 2	Bit ON in caso di perdita dei dati a ritenzione dell'area EEPROM.	R/W
	Bit 4	Bit ON in caso di reset della CPU o intervento del watch-dog.	R/W
	Bit 5	Bit ON in caso di overflow dello stack sull'area riservata alla RAM.	R/W
	Bit 7	Bit ON in caso di errore/guasto dell'EEPROM.	R/W
	Bit 8	Bit ON in caso di errore/guasto dell'orologio.	R/W
	Bit 10	Bit ON in caso di overflow dello stack	R/W

		delle interrupt a tempo.	
SM2	1002	Indirizzo dispositivo	
		Indirizzo (word) di protocollo ModBus del dispositivo. All'accensione, se SM1.0 = 1 , il valore viene inizializzato a 1, altrimenti viene mantenuto il dato precedentemente salvato.	R/W
SM3	1003	Tempo ciclo	
		Tempo dell'ultimo ciclo di scansione del programma (risoluzione 100 µS).	R
SM4	1004	Minimo tempo ciclo	
		Tempo minimo del ciclo di scansione del programma rilevato (risoluzione 100 µS).	R
SM5	1005	Massimo tempo ciclo	
		Tempo massimo del ciclo di scansione del programma rilevato (risoluzione 100 µS).	R
SM6	1006	Intervallo dell'interrupt a tempo n°1	
SM7	1007	Intervallo dell'interrupt a tempo n°2	
		Word che definiscono l'intervallo degli interrupt a tempo. Il valore può essere impostato tra 1 e 100 ms (esempio: SM6=1 → 1 ms, SM6=100 → 100 ms). Per valori di SM6 e SM7 non compresi tra 1 e 100, l'interrupt corrispondente viene fissato di default a 100 ms. All'accensione sono entrambi fissati a 100 → 100 ms. Nel codice ladder delle due interrupt, non è consentito l'utilizzo di funzioni che accedano all'area EEPROM ed all'area MMC.	R/W
SM8	1008	Contrasto LCD	
		Contrasto display LCD 0...100 → 0...100%. All'accensione, se SM1.0 = 1 , il valore viene inizializzato a 50 → 50%, altrimenti viene mantenuto il dato precedentemente salvato.	R/W
SM9	1009	Tempo minimo lampada	
		Retroilluminazione display LCD 0...1000 → 0...1000minuti, 0 → sempre acceso.	R/W

		All'accensione, se SM1.0 = 1 , il valore viene inizializzato a 0 → sempre acceso, altrimenti viene mantenuto il dato precedentemente salvato.	
SM10	1010	Touch screen X	
SM11	1011	Touch screen Y	
		Coordinate del punto di contatto al display LCD (X = 0...319, Y = 0...239) X=0, Y=0 → angolo superiore sinistro Quando il display non viene toccato, X = 500, Y = 500	R
SM12	1012	Touch screen FLAGS	
		Bit 0 Bit ON in caso di evento: botton up, botton down o autorepeat.	R
		Bit 1 Bit ON in caso di touch down (pressione sul display).	R
		Bit 2 Bit ON in caso touch up (rilascio della pressione sul display).	R
		Bit 3 Bit ON in caso di touch press (pressione continua sul display).	R
		Bit 4 Bit ON in caso di touch repeat (evento autorepeat).	R
SM13	1013	Lingua	
		Il numero delle lingue per i messaggi di testo della grafica è fissato dal <i>TdDesigner</i> . Questa word definisce in quale lingua sono correntemente visualizzati i messaggi di testo (se n è il numero di lingue fissato dal TdDesigner, SM13 può variare da 0 a n-1). All'accensione, se SM1.0 = 1 , il valore viene inizializzato a 0 → prima lingua, altrimenti viene mantenuta la lingua selezionata.	R/W
SM14	1014	Numero pagina visualizzata	
		Word che indica il numero della pagina correntemente visualizzata (default 1, all'accensione viene sempre visualizzata la prima pagina).	R

SM15	1015	Numero pagina da visualizzare	
		<p>Word che indica il numero della pagina da visualizzare.</p> <p>La scrittura in questa word di un numero corrispondente ad una pagina fisicamente creata dal TdDesigner comporterà un salto immediato a tale pagina, altrimenti la pagina visualizzata resterà quella di prima.</p> <p>Dopo il cambio pagina, la word viene automaticamente portata a 0.</p> <p>All'accensione, se SM1.0 = 1, il valore viene inizializzato a 0 → nessun cambio pagina, altrimenti viene mantenuta la pagina precedentemente scelta.</p>	R/W

SM N°	ModBus Address	Descrizione / Significato	
SM16	1016	Area ultima variabile modificata	
		Word che indica (per un solo ciclo di scansione) l'indice corrispondente all'ultima area di memoria salvata dalla grafica. Nel dettaglio gli indici corrispondenti alle aree	R
		Area word V → 1 Area word SM → 2 Area word AI → 3 Area word TR → 4 Area word AQ → 5 Area word I → 6 Area word Q → 7 Area word T → 8 Area word PT → 9 Area word C → 10 Area word PV → 11 Area double V → 12 Area double SM → 13 Area word M → 14 Area word EEPROM → 15 Area word MMC → 16 Area byte TX COM1 → 17 Area byte RX COM1 → 18 Area byte TX EXP1 → 19 Area byte RX EXP1 → 20 Area byte TX COM2 → 21 Area byte RX COM2 → 22	
SM17	1017	Numero ultima variabile modificata	
		Word che indica (per un solo ciclo di scansione) il numero dell'ultima area di memoria salvata dalla grafica. Ad esempio, se la grafica modifica la variabile VW30 , si avrà, per il solo ciclo di	R

		scansione successivo alla modifica, SM16 = 1 e SM17 = 30 . Nel ciclo successivo le due aree verranno automaticamente poste a 0.	
SM18	1018	Tempo attivazione buzzer (x10ms)	
		Tempo di attivazione buzzer del dispositivo in multipli di 10ms. Il valore di default è 0xFFFF = 65536 = buzzer spento, valore caricato anche al termine dell'attivazione. Con SM18 = 0, il buzzer si spegnerà solo al tocco del display.	R/W
SM19	1019	Uscite digitali TD320	
		Bit 0 Bit ON → uscita relè attiva (tra morsetti 1 e 2 tensione +V)	R/W
SM20	1020	Percentuale CPU grafica	
		Percentuale del tempo impiegato per eseguire le istruzioni relative alla grafica. Valori possibili 10...90 → 10...90%, default 50 → 50% (metà tempo alla grafica e metà al PLC).	R/W
SM21	1021	Percentuale CPU grafica cambio pagina	
		Percentuale del tempo impiegato per eseguire le istruzioni relative alla grafica solo durante un cambio di pagina. Una volta eseguito, la gestione effettiva del tempo è decisa da SM20. Valori possibili 10...90 → 10...90%, default 50 → 50%	R/W
SM30	1030	Secondi	
		Secondi orologio interno (0...59)	R/W
SM31	1031	Minuti	
		Minuti orologio interno (0...59)	R/W
SM32	1032	Ore	
		Ore orologio interno (0...23)	R/W
SM33	1033	Giorno	
		Giorno orologio interno (1...31)	R/W
SM34	1034	Mese	
		Mese orologio interno (1...12)	R/W

SM35	1035	Anno	
		Anno orologio interno (0...99)	R/W
SM36	1036	Giorno della settimana	
		Giorno settimana orologio interno (0 → Domenica, 6 → Sabato)	R/W
SM37	1037	Filtro ingressi digitali	
		<p>E' possibile filtrare i segnali degli ingressi digitali impostando un tempo di ritardo. Se lo stato dell'ingresso cambia, il nuovo stato verrà accettato solo se l'ingresso lo manterrà per il tempo impostato. I dati verranno accettati dopo che il filtro avrà eliminato i disturbi e fissato le linee degli ingressi su valori stabili.</p> <p>Il TD320 supporta filtri con tempi di ritardo compresi tra 0 e 50ms (default 10ms)</p>	R/W
SM38	1038	Ingressi digitali	
		Bit 0	Bit ON → Ingresso digitale DI1 attivo
		Bit 1	Bit ON → Ingresso digitale DI2 attivo
		Bit 2	Bit ON → Ingresso digitale DI3 attivo
		Bit 3	Bit ON → Ingresso digitale DI4 attivo
		Bit 4	Bit ON → Ingresso digitale DI5 attivo
		Bit 5	Bit ON → Ingresso digitale DI6 attivo
		Bit 6	Bit ON → Ingresso digitale DI7 attivo
		Bit 7	Bit ON → Ingresso digitale DI8 attivo
SM39	1039	Conf. COM1 in modalità Free-port	
SM40	1040	Conf. EXP1 in modalità Free-port	
SM41	1041	Conf. COM2 in modalità Free-port	
		<p>Word che abilitano il funzionamento delle porte seriali in modalità free-port e ne impostano i parametri.</p> <p>Abilitando tale modalità, il protocollo di comunicazione che utilizza la seriale viene disabilitato, consentendo l'accesso diretto alle funzione di trasmissione e ricezione dei dati sulla porta. Questi parametri sono inizializzati all'accensione a 0 (modalità free-</p>	R/W

		port disabilitata)	
		Bit 0÷3 Questi bit impostano la velocità di comunicazione della porta seriale nella modalità free-port secondo i seguenti valori (baud): <div> <div>0 → 110</div> <div>6 → 4800</div> <div>1 → 150</div> <div>7 → 9600</div> <div>2 → 300</div> <div>8 → 19200</div> <div>3 → 600</div> <div>9 → 28800</div> <div>4 → 1200</div> <div>10 → 38400</div> <div>5 → 2400</div> <div>11 → 57600</div> </div>	R/W
		Bit 4÷7 Questi bit impostano il formato dei dati di comunicazione della porta seriale nella modalità free-port: 7,8 = numero bit di dato, N = nessun controllo sulla parità, O = parità dispari, E = parità pari, 1,2 = numero bit di stop <div> <div>0 → 8, N, 1</div> <div>6 → 8, N, 2</div> <div>1 → 8, O, 1</div> <div>7 → 8, O, 2</div> <div>2 → 8, E, 1</div> <div>8 → 8, E, 2</div> <div>3 → 7, N, 1</div> <div>9 → 7, N, 2</div> <div>4 → 7, O, 1</div> <div>10 → 7, O, 2</div> <div>5 → 7, E, 1</div> <div>11 → 7, E, 2</div> </div>	R/W
		Bit 8 Bit impostato a “1” abilita la modalità free-port. “0” riporta la gestione della seriale al protocollo selezionato nella fase di programmazione.	R/W
SM42	1042	Num. Byte in buffer ricezione COM1	
SM43	1043	Num. Byte in buffer ricezione EXP1	
SM44	1044	Num. Byte in buffer ricezione COM2	
		Word contenenti, per ciascuna seriale, il numero di caratteri validi presenti nel buffer di ricezione. Utili in modalità free-port per controllare il numero di caratteri ricevuti. Qualsiasi scrittura imposta il valore corrispondente a zero, cioè svuota il buffer di	R/W

		ricezione.												
SM45	1045	Baud rate seriale COM1 (default 9600baud)												
SM49	1049	Baud rate seriale EXP1 (default 9600baud)												
SM53	1053	Baud rate seriale COM2 (default 57600baud)												
		<p>Il valore impostato definisce la velocità di comunicazione della porta seriale per il protocollo ModBus, se abilitato (baud).</p> <p>N.B. Perché le modifiche siano attive, è necessario impostare queste word nel codice di inizializzazione. Nel caso non venga effettuata nessuna modifica oppure venga effettuata in altre parti del programma, il baud rate rimarrà quello impostato di default all'accensione.</p>	R/W											
		<table><tr><td>0 → 110</td><td>6 → 4800</td></tr><tr><td>1 → 150</td><td>7 → 9600</td></tr><tr><td>2 → 300</td><td>8 → 19200</td></tr><tr><td>3 → 600</td><td>9 → 28800</td></tr><tr><td>4 → 1200</td><td>10 → 38400</td></tr><tr><td>5 → 2400</td><td>11 → 57600</td></tr></table>		0 → 110	6 → 4800	1 → 150	7 → 9600	2 → 300	8 → 19200	3 → 600	9 → 28800	4 → 1200	10 → 38400	5 → 2400
0 → 110	6 → 4800													
1 → 150	7 → 9600													
2 → 300	8 → 19200													
3 → 600	9 → 28800													
4 → 1200	10 → 38400													
5 → 2400	11 → 57600													
SM46	1046	Formato seriale COM1 (default 8, N, 1)												
SM50	1050	Formato seriale EXP1 (default 8, N, 1)												
SM54	1054	Formato seriale COM2 (8,N,1 non modificabile)												
		<p>Il valore impostato definisce il formato dei dati di comunicazione della porta seriale per il protocollo ModBus, se abilitato.</p> <p>N.B. Perché le impostazioni siano attive, è necessario modificare questa word nel codice di inizializzazione. Nel caso non venga effettuata nessuna modifica oppure venga effettuata in altre parti del programma, il formato rimarrà quello impostato di default all'accensione.</p>	R/W											

		0 → 8, N, 1 6 → 8, N, 2 1 → 8, O, 1 7 → 8, O, 2 2 → 8, E, 1 8 → 8, E, 2 3 → 7, N, 1 9 → 7, N, 2 4 → 7, O, 1 10 → 7, O, 2 5 → 7, E, 1 11 → 7, E, 2	
SM47	1047	Ritardo RX/TX COM1 (default 20ms)	
SM51	1051	Ritardo RX/TX EXP1 (default 100ms)	
SM55	1055	Ritardo RX/TX COM2 (default 1ms)	
		Il valore impostato in ms definisce: <ul style="list-style-type: none"> • Protocollo slave: il ritardo minimo tra la fine della ricezione seriale di dati provenienti da un dispositivo master, all'inizio della trasmissione dei dati della risposta del TD320 (max 100ms). • Protocollo master: l'attesa massima tra l'inizio della trasmissione dei dati dell'interrogazione da parte del TD320, alla ricezione completa dei dati della risposta di un dispositivo slave. 	R/W
SM48	1048	Num. errori per segnalazione COM1	
SM52	1052	Num. errori per segnalazione EXP1	
SM56	1056	Num. errori per segnalazione COM2	
		Il valore impostato in questa word definisce il numero di errori di comunicazione consecutivi dopo i quali viene segnalata l'anomalia nei rispettivi bit delle word " Stato seriale ". Il valore di default per tutte le porte è 10.	R/W
SM57	1057	Stato seriale COM1 1-16	
SM58	1058	Stato seriale COM1 17-32	
SM59	1059	Stato seriale COM1 33-48	
SM60	1060	Stato seriale COM1 49-64	
SM61	1061	Stato seriale COM1 65-80	
SM62	1062	Stato seriale COM1 81-96	
SM63	1063	Stato seriale COM1 97-112	

SM64	1064	Stato seriale COM1 113-128	
SM65	1065	Stato seriale COM1 129-144	
SM66	1066	Stato seriale COM1 145-160	
SM67	1067	Stato seriale COM1 161-176	
SM68	1068	Stato seriale COM1 177-192	
SM69	1069	Stato seriale COM1 193-208	
SM70	1070	Stato seriale COM1 209-224	
SM71	1071	Stato seriale COM1 225-240	
SM72	1072	Stato seriale COM1 241-256	
		<p>Queste word contengono lo stato della seriale di comunicazione COM1.</p> <p>Ciascun bit di ogni word segnala una condizione di mancata comunicazione (fuori linea) o errore per ciascuno dei dati trasmessi o ricevuti mediante le istruzioni COM_1÷256 (ad esempio SM66.9=1 indica errore nella istruzione numero COM_154(...)). Nel caso di seriale impostata con protocollo slave, la condizione di errore viene segnalata ponendo a “1” tutti i bit della word SM57.</p> <p>All'accensione tutte le word vengono inizializzate a 0.</p>	R
SM73	1073	Stato seriale EXP1 1-16	
SM74	1074	Stato seriale EXP1 17-32	
SM75	1075	Stato seriale EXP1 33-48	
SM76	1076	Stato seriale EXP1 49-64	
SM77	1077	Stato seriale EXP1 65-80	
SM78	1078	Stato seriale EXP1 81-96	
SM79	1079	Stato seriale EXP1 97-112	
SM80	1080	Stato seriale EXP1 113-128	
SM81	1081	Stato seriale EXP1 129-144	
SM82	1082	Stato seriale EXP1 145-160	
SM83	1083	Stato seriale EXP1 161-176	
SM84	1084	Stato seriale EXP1 177-192	
SM85	1085	Stato seriale EXP1 193-208	
SM86	1086	Stato seriale EXP1 209-224	

SM87	1087	Stato seriale EXP1 225-240	
SM88	1088	Stato seriale EXP1 241-256	
		<p>Queste word contengono lo stato della seriale di comunicazione EXP1.</p> <p>Ciascun bit di ogni word segnala una condizione di mancata comunicazione (fuori linea) o errore per ciascuno dei dati trasmessi o ricevuti mediante le istruzioni EXP_1÷256 (ad esempio SM80.4=1 indica errore nella istruzione numero EXP_117(...)).</p> <p>Nel caso di seriale impostata con protocollo slave, la condizione di errore viene segnalata ponendo a “1” tutti i bit della word SM73.</p> <p>All'accensione tutte le word vengono inizializzate a 0.</p>	R
SM89	1089	Stato seriale COM2 1-16	
SM90	1090	Stato seriale COM2 17-32	
SM91	1091	Stato seriale COM2 33-48	
SM92	1092	Stato seriale COM2 49-64	
SM93	1093	Stato seriale COM2 65-80	
SM94	1094	Stato seriale COM2 81-96	
SM95	1095	Stato seriale COM2 97-112	
SM96	1096	Stato seriale COM2 113-128	
SM97	1097	Stato seriale COM2 129-144	
SM98	1098	Stato seriale COM2 145-160	
SM99	1099	Stato seriale COM2 161-176	
SM100	1100	Stato seriale COM2 177-192	
SM101	1101	Stato seriale COM2 193-208	
SM102	1102	Stato seriale COM2 209-224	
SM103	1103	Stato seriale COM2 225-240	
SM104	1104	Stato seriale COM2 241-256	

		<p>Queste word contengono lo stato della seriale di comunicazione COM2.</p> <p>Dal momento che questa porta può essere configurata solo come ModBus slave, la condizione di errore viene segnalata ponendo a “1” tutti i bit della word SM89.</p> <p>All'accensione tutte le word vengono inizializzate a 0.</p>	R
SM105	1105	Numero time-out COM1	
SM107	1107	Numero time-out EXP1	
SM109	1109	Numero time-out COM2	
		<p>Se per la porta corrispondente è fissato un protocollo Master, indica il numero di pacchetti di informazione non ricevuti durante la comunicazione.</p> <p>All'accensione i contatori vengono inizializzati a 0.</p>	R
SM106	1106	Numero errori COM1	
SM108	1108	Numero errori EXP1	
SM110	1110	Numero errori COM2	
		<p>Se per la porta corrispondente è fissato un protocollo Master, indica il numero di pacchetti di informazione errati durante la comunicazione.</p> <p>All'accensione i contatori vengono inizializzati a 0.</p>	R
SM111	1111	Ritardo minimo nuova trasmissione COM1[#]	
SM112	1112	Ritardo minimo nuova trasmissione EXP1[#]	
SM113	1113	Ritardo minimo nuova trasmissione COM2[#]	

[#] Disponibili dalla versione firmware 1.12

	Se per la porta corrispondente è fissato un protocollo Master, fissa il ritardo minimo per una nuova trasmissione dopo la risposta di un dispositivo slave. Valori possibili 0...100 → 0...100ms, default 5 → 5ms	R/W
--	--	------------

5.3 Area Ingressi Digitali I

L'area di memoria I è composta da **32 word** e può essere utilizzata per contenere lo stato degli ingressi digitali letti tramite seriale da altri dispositivi.

E' organizzata in word: ciascuno dei 16 bit di una word può rappresentare lo stato di un ingresso. E' accessibile anche a bit, in modo da poter gestire ogni singolo ingresso.

5.4 Area Uscite Digitali Q

L'area di memoria Q è composta da **32 word** e può essere utilizzata per contenere lo stato delle uscite per poi scriverle tramite seriale su altri dispositivi.

E' organizzata in word: ciascuno dei 16 bit di una word può rappresentare lo stato di un'uscita. E' accessibile anche a bit, in modo da poter gestire ogni singola uscita.

5.5 Area Marker M

L'area di memoria M è composta da **50 word** e contiene lo stato di tutti i marker (contatti bit) utilizzati nel programma.

E' organizzata in word: ciascuno dei 16 bit di una word rappresenta lo stato di un marker. Ad esempio, lo stato del marker M5 è memorizzato nel bit n°4 della word 1 dell'area M. Il marker M5 è quindi accessibile come M1.4 (contatto bit su word), ma anche come singolo bit M5 (contatto o bobina).

5.6 Area Ingressi Analogici AI

L'area di memoria AI è composta da **32 word** e può essere utilizzata per contenere lo stato degli ingressi analogici letti tramite seriale da altri dispositivi.

E' organizzata in word: ciascuna può rappresentare lo stato di un ingresso analogico.

5.7 Area Uscite Analogiche AQ

L'area di memoria AQ è composta da **32 word** e può essere utilizzata per contenere lo stato delle uscite analogiche lette tramite seriale da altri dispositivi.

E' organizzata in word: ciascuna può rappresentare lo stato di un'uscita analogica.

5.8 Aree Timer T e Preset Timer PT

L'area di memoria timer T è composta da **128 word**. Se il timer è abilitato, la variazione del contenuto dell'area di memoria è regolata dal tipo di timer, impostato al momento dell'attivazione.

L'area di memoria preset timer PT è formata da **128 word** e contiene i valori di attivazione dei contatti (preset) dei rispettivi timer.

Le aree sono organizzate a word, quindi la risoluzione dei timer e dei preset è 16bit (32767).

5.9 Area Contatori C e Preset Contatori PV

L'area di memoria contatori C è composta da **64 word**. Se il contatore è abilitato, la variazione del contenuto dell'area di memoria è regolata dal tipo di contatore.

L'area di memoria preset contatori PV è formata da **64 word** e contiene i valori di attivazione dei contatti (preset) dei rispettivi contatori.

Le aree sono organizzate a word, quindi la risoluzione dei contatori e dei preset è 16bit (da -32768 a 32767).

5.10 Area Relè Bistabili B

L'area di memoria relè bistabili B è composta da **128 bit**. E' organizzata a bit, quindi ogni relè bistabile è individuato da un singolo bit.

5.11 Area EEPROM

L'area di memoria EEPROM è composta da **1000 word**. E' la memoria a ritenzione dove possono essere salvati i dati che necessitano di essere mantenuti anche se il TD320 dovesse rimanere spento per periodi molto lunghi (oltre i 6 mesi). I dati salvati in quest'area sono infatti testati all'accensione per verificare la loro integrità, e le anomalie, segnalate attivando il bit **SM1.2**, comportano l'inizializzazione di tutta l'area a 0.

L'accesso e la scrittura a quest' area (1000000 le scritture garantite per ogni cella) richiedono un tempo nettamente superiore a qualsiasi altra (30/40ms), quindi è consigliabile non utilizzarla per accessi continuativi, ma solamente per copiare all'accensione i dati in essa contenuti per esempio nell'area V e poi utilizzare quest'ultimi per un accesso più rapido (5/10µs).

5.12 Area MMC

L'area di memoria MMC è composta da **30000 word**. E' la memoria a ritenzione dove possono essere salvati grandi quantità di dati da mantenere anche in assenza di alimentazione.

La memoria è di tipo EEPROM. L'accesso risulta quindi più lento rispetto alle aree V e SM, ed il TD320 non esegue alcun controllo sull'integrità dei dati memorizzati in tale area.

5.13 Area TX/RX COM1

L'area di memoria TX/RX COM1 è composta da **200 byte**. È l'area di memoria utilizzata per gestire i dati in transito nella porta seriale COM1.

I primi 100 byte (TX-0...TX-99) sono utilizzati per caricare i dati da trasmettere, gli ultimi 100 byte (RX-0...RX-99) sono utilizzati per salvare i dati ricevuti dalla porta seriale COM1.

Il loro utilizzo trova significato solo nella modalità free-port, mentre nella modalità normale sono gestiti direttamente dal protocollo selezionato in fase di programmazione.

5.14 Area TX/RX EXP1

L'area di memoria TX/RX EXP1 è composta da **200 byte**. È l'area di memoria utilizzata per gestire i dati in transito nella porta seriale EXP1.

I primi 100 byte (TX-0...TX-99) sono utilizzati per caricare i dati da trasmettere, gli ultimi 100 byte (RX-0...RX-99) sono utilizzati per salvare i dati ricevuti dalla porta seriale EXP1.

Il loro utilizzo trova significato solo nella modalità free-port, mentre nella modalità normale sono gestiti direttamente dal protocollo selezionato in fase di programmazione.

6 Protocolli di comunicazione

Il TD320 può comunicare con tutti i dispositivi che supportano i seguenti protocolli seriali:

- ModBus RTU
- Nais Matsushita master

Il terminale ha 3 porte di comunicazione seriale (COM1, EXP, COM2), analizzate da un punto di vista elettrico nel cap.3. Le porte sono tuttavia gestite in maniera diversa tra loro e vanno analizzate separatamente.

6.1 Gestione porte di comunicazione

La comunicazione tra il TD320 ed altri dispositivi è gestita dalla parte PLC del terminale, quindi la configurazione delle porte e le istruzioni devono essere implementate nell'ambiente di sviluppo PLProg 4.xx.

Generalmente le bobine del diagramma Ladder vengono eseguite seguendo l'ordine sequenziale di scrittura del diagramma stesso. L'istruzione relativa alla bobina alla riga n+1_esima non viene eseguita finchè non è stata completamente eseguita l'istruzione relativa alla bobina alla riga n_esima (per bobine posizionate nella stessa colonna).

La gestione della trasmissione e della ricezione dei dati è invece **asincrona** rispetto al ciclo di esecuzione del codice Ladder.

Quando deve essere eseguita un'istruzione di lettura/scrittura da un dispositivo (riga n_esima), il controllo passa subito all'istruzione successiva (riga n+1_esima), senza attendere che i dati siano stati effettivamente letti/scritti.

Il trasferimento effettivo dei dati nella linea seriale viene eseguito in maniera indipendente dalla normale scansione del codice Ladder, in tempi diversi a seconda della porta utilizzata.

6.1.1 Porte COM1 ed EXP1

Le porte COM1 ed EXP1 possono essere configurate con protocollo ModBus (master o slave), Nais Matsushita master, Control Technique. Sono le porte solitamente usate per la comunicazione con altri dispositivi (PLC, ecc.).

Il controllo sulla comunicazione viene effettuato ogni 1ms. Ciò significa che il flusso dei dati nella seriale corrispondente sarà controllato 1000 volte al secondo.

6.1.2 Porta COM2

La porta COM2 può essere configurata solo con protocollo ModBus slave. E' la porta usata per la programmazione del terminale da PC.

Il controllo sulla comunicazione viene effettuato ogni ciclo di scansione del codice Ladder. Ciò significa che il flusso dei dati nella seriale COM2 sarà controllato 1 volta alla fine di ogni ciclo di scansione.

6.2 Protocollo ModBus RTU

Il ModBus su linea seriale è un protocollo Master-Slave. In una rete di questo tipo c'è un solo nodo (il Master) che interroga e comanda gli Slave ed elabora le risposte. I nodi Slave tipicamente non trasmettono dati senza una specifica richiesta del Master e non comunicano direttamente tra di loro.

Un dispositivo nella linea seriale (un nodo della rete) è univocamente determinato da un identification number (ID, variabile da 1 a 255), chiamato indirizzo ModBus Slave: due dispositivi non possono avere lo stesso indirizzo.

I destinatari della richiesta (uno o più nodi Slave) sono individuati dal Master tramite il loro ID, quindi i dati che transitano nella linea hanno una destinazione ben precisa.

Il Master è il responsabile della linea: non ha indirizzo ID specifico e può leggere o scrivere dati sotto forma di word o bit in uno o più dispositivi Slave, specificando l'ID di destinazione.

I dati letti o scritti vengono salvati nel dispositivo di destinazione in registri identificati da uno specifico ModBus address (variabile da 1 a 65535). Ad ogni ModBus address può corrispondere un registro (area word di memoria) oppure il singolo bit di un registro (particolare bit di un'area di memoria).

Si riporta nella figura seguente la lista delle operazioni possibili in una comunicazione ModBus: lettura e scrittura di word o bit singole o multiple.

Caratteristiche protocollo Modbus RTU		
Baud-rate	Programmabile	
Formato	8,N,1 (8 bit, no parità, 1 stop) (default)	
Funzioni supportate	BITS READING	(0x01, 0x02)
	WORDS READING (max 20 word)	(0x03, 0x04)
	SINGLE BIT WRITING	(0x05)
	SINGLE WORD WRITING	(0x06)
	MULTIPLE BITS WRITING	(0x0F)
	MULTIPLE WORDS WRITING (max 20 word)	(0x10)
Codici di errore	ILLEGAL FUNCTION CODE	(0x01)
	ILLEGAL DATA ADDRESS	(0x02)
	ILLEGAL DATA VALUE	(0x04)
Broadcast	Scrittura simultanea a tutti gli slave collegati usando l'indirizzo 0x00 e senza nessuna risposta da parte degli slave.	
Interrogazione con indirizzo slave sconosciuto	Interrogazione con indirizzo 0xFF a cui risponde qualsiasi slave collegato.	

6.2.1 ModBus RTU Master

Il protocollo ModBus master può essere configurato solo per le porte COM1 ed EXP1.

Con questa configurazione il TD320 avrà il controllo del transito dei dati nella porta corrispondente. Per ognuna delle due porte possono essere attivi fino a 256 frame (pacchetti attivi) contemporaneamente.

Ciascun frame corrisponde ad un'istruzione di comunicazione diretta:

- **Lettura da uno Slave:** Lettura dallo slave dell'indirizzo ModBus corrispondente al dato (ai dati) di interesse e memorizzazione nei registri del master. Ogni istruzione può leggere fino a 16 word consecutive.
- **Scrittura su uno Slave:** Scrittura del dato (dei dati) di interesse del master nello slave all'indirizzo ModBus corrispondente ai dati da sovrascrivere. Ogni istruzione può scrivere fino a 16 word consecutive.
- **Lettura/scrittura su uno Slave:** Normalmente si va a leggere dallo slave il dato di interesse e lo si salva nel master. Quando il dato interno al TD320 varia per effetto del programma, si va a scrivere il dato modificato nello slave. Ogni istruzione di lettura/scrittura può operare solo su 1 word.

6.2.2 ModBus RTU Slave

Il protocollo ModBus slave può essere configurato per tutte e tre le porte COM1, EXP1 e COM2.

Con questa configurazione tutte le risorse del terminale sono a disposizione del dispositivo master eventualmente collegato.

Le tabelle seguenti indicano tutti i dati (word e bit) accessibili attraverso il protocollo Modbus. A ciascuna area di memoria corrisponde un indirizzo ModBus distinto (per l'accesso a word o a bit), variabile da 0 a 65536.

Per ciascuno vengono riportati gli accessi in lettura e scrittura e il valore assunto all'accensione del TD320. A seconda del valore di inizializzazione, si distinguono i seguenti casi:

1. **“ROM”** valore fisso definito dal programma.
2. **“EEP”** valore memorizzato in memoria EEPROM, mantenuto per almeno 10 anni anche in mancanza di alimentazione.
3. **“TAMP”** valore memorizzato in memoria RAM con batteria tampone. Anche questi dati vengono mantenuti in mancanza di alimentazione, ma per un tempo limitato (4/6 mesi circa).
4. **“VALORE DEFINITO”** il valore assunto dal dato all'accensione corrisponde al valore indicato nella tabella.

ACCESSO A WORD			
MODBUS ADDRESS	DESCRIZIONE	READ/ WRITE	RESET VALUE
0	Tipo di dispositivo	R	ROM
1	Versione Firmware	R	ROM
2	Protocollo attivato su COM1	R	ROM
3	Protocollo attivato su EXP1	R	ROM
4	Protocollo attivato su COM2	R	ROM
5	Indirizzo di protocollo	R	TAMP
6	Versione BOOT	R	ROM
10	Secondi orologio TD320	R/W	TAMP
1000 ÷ 1199	Word area special marker SM	R/W	TAMP
2000 ÷ 2999	Word area variabili V	R/W	TAMP
12000 ÷ 12127	Word area timer T	R/W	0
13000 ÷ 13127	Word area preset timer PT	R/W	0
14000 ÷ 14063	Word area contatori C	R/W	0
15000 ÷ 15063	Word area preset contatori PV	R/W	0
16000 ÷ 16099	Word area buffer TX COM1	R	0
16500 ÷ 16599	Word area buffer RX COM1	R	0
17000 ÷ 17099	Word area buffer TX EXP1	R	0
17500 ÷ 17599	Word area buffer RX EXP1	R	0
18000 ÷ 18099	Word area buffer TX COM2	R	0
18500 ÷ 18599	Word area buffer RX COM2	R	0

19000 ÷ 19031	Word area ingressi analogici AI	R	0
19200 ÷ 19215	Word area trimmer TR	R	0
19400 ÷ 19431	Word area uscite analogiche AQ	R	0
19800 ÷ 19927	Word percentuali prop/integ/deriv/uscite PID		
19800	% Azione proporzionale PID1	R	0
19801	% Azione integrale PID1	R	TAMP
19802	% Azione derivativa PID1	R	TAMP
19803	% Uscita PID1	R	TAMP
19804	% Azione proporzionale PID2	R	0
.....		
19927	% Uscita PID128	R	TAMP
20000 ÷ 20999	Word area EEProm	R/W	EEP
30000 ÷ 59999	Word area MMC	R/W	EEP

ACCESSO A WORD			
MODBUS ADDRESS	DESCRIZIONE	READ/ WRITE	RESET VALUE
90	Contatti n.a. posizionatori POS1÷POS16	R	0
95	Contatti n.a. tuning posizionatori POS1÷POS16	R	0
100	Contatti n.a. ingressi digitali I1 ÷ I16	R	0
101	Contatti n.a. ingressi digitali I17 ÷ I32	R	0
.....		
131	Contatti n.a. ingressi digitali I497 ÷ I512	R	0
150	Contatti n.a. uscite digitali Q1 ÷ Q16	R	0
151	Contatti n.a. uscite digitali Q17 ÷ Q32	R	0
.....		
181	Contatti n.a. uscite digitali Q497 ÷ Q512	R	0
200	Contatti n.a. relè bistabili B1 ÷ B16	R/W	0
201	Contatti n.a. relè bistabili B17 ÷ B32	R/W	0
.....		
207	Contatti n.a. relè bistabili B113 ÷ B128	R/W	0
250	Contatti n.a. marker M1 ÷ M16	R	0

251	Contatti n.a. marker M17 ÷ M32	R	0
.....		
299	Contatti n.a. marker M785 ÷ M800	R	0
300	Contatti n.a. timer T1 ÷ T16	R	0
301	Contatti n.a. timer T17 ÷ T32	R	0
.....		
307	Contatti n.a. timer T113 ÷ T128	R	0
350	Contatti n.a. contatori C1 ÷ C16	R	0
351	Contatti n.a. contatori C17 ÷ C32	R	0
352	Contatti n.a. contatori C33 ÷ C48	R	0
353	Contatti n.a. contatori C49 ÷ C64	R	0

ACCESSO A BIT			
MODBUS ADDRESS	DESCRIZIONE	READ/ WRITE	RESET VALUE
1440	Contatto n.a. Posizionatore POS1	R	0
1441	Contatto n.a. Posizionatore POS2	R	0
.....		
1455	Contatto n.a. Posizionatore POS15	R	0
1520	Contatto n.a. tuning posizion. POS1	R	0
1521	Contatto n.a. tuning posizion. POS2	R	0
.....		
1535	Contatto n.a. tuning posizion. POS15	R	0
1600	Contatto n.a. ingresso digitale I1	R/W	0
1601	Contatto n.a. ingresso digitale I2	R/W	0
.....		
2111	Contatto n.a. ingresso digitale I512	R/W	0
2400	Contatto n.a. uscita digitale Q1	R/W	0
2401	Contatto n.a. uscita digitale Q2	R/W	0
.....		
2911	Contatto n.a. uscita digitale Q512	R/W	0
3200	Contatto n.a. relè bistabile B1	R/W	0
3201	Contatto n.a. relè bistabile B2	R/W	0
.....		
3327	Contatto n.a. relè bistabile B128	R/W	0

4000	Contatto n.a. marker M1	R/W	0
4001	Contatto n.a. marker M2	R/W	0
.....		
4799	Contatto n.a. marker M800	R/W	0
4800	Contatto n.a. timer T1	R	0
4801	Contatto n.a. timer T2	R	0
.....		
4927	Contatto n.a. timer T128	R	0
5600	Contatto n.a. contatore C1	R	0
5601	Contatto n.a. contatore C2	R	0
.....		
5663	Contatto n.a. contatore C64	R	0
16000	Bit 0 area special marker SM0	R/W	TAMP
16001	Bit 1 area special marker SM0	R/W	TAMP
.....		
19199	Bit 15 area special marker SM199	R/W	TAMP
32000	Bit 0 area variabili V0	R/W	TAMP
32001	Bit 1 area variabili V0	R/W	TAMP
.....		
63999	Bit 15 area variabili V2000	R/W	TAMP

6.3 Protocollo NAIS Matsushita Master

E' il protocollo che permette la lettura e la scrittura di dati (bit o word) sui PLC Nais Matsushita.

Generalmente l'interfaccia di comunicazione è RS232, la velocità è 9600baud (bit/sec), il formato di comunicazione 8,O,1 (8 bit di dato, parità dispari, 1 bit di stop).

Le tabelle seguenti indicano tutti gli elementi che possono essere letti/scritti dal PLC. L'indirizzo del bit o della word da leggere o scrivere si ottiene sommando l'indirizzo reale del bit/word (compreso tra Min e Max) al valore indicato nella colonna Offset. Ogni istruzione "COM" o "EXP" può eseguire la lettura o scrittura di più dati consecutivi; il numero massimo per ciascun tipo di dato è indicato nella colonna "Max numero bit/word letti/scritti consecutivi".

ACCESSO A BIT						
CONTATTO	NOTAZ.	MIN	MAX	OFFSET	READ/ WRITE	MAX NUMERO BIT LETTI / SCRITTI CONSECUTIVI
EXTERNAL INPUT	X	0	9999	0	R	8
EXTERNAL OUTPUT	Y	0	9999	10000	R/W	8
INTERNAL RELAY	R	0	9999	20000	R/W	8
LINK RELAY	L	0	9999	30000	R/W	8
TIMER	T	0	9999	40000	R	8
COUNTER	C	0	9999	50000	R	8

ACCESSO A WORD						
CODICE WORD	NOTAZ.	MIN	MAX	OFFSET	READ/ WRITE	MAX NUMERO WORD LETTE / SCRITTE CONSECUTIVE
EXTERNAL INPUT	X	0	999	0	R	10
EXTERNAL OUTPUT	Y	0	999	1000	R/W	10(R) / 7 (W)
INTERNAL RELAY	R	0	999	2000	R/W	10(R) / 7 (W)
LINK RELAY	L	0	999	3000	R/W	10(R) / 7 (W)
TIMER	T	0	999	4000	R	10
COUNTER	C	0	999	5000	R	10
INDEX REG. X		0	0	6000	R/W	1
INDEX REG. Y		0	0	6001	R/W	1
INDEX REG. D		0	0	6002	R/W	1

DATA REGISTER	DT	0	9999	10000	R/W	10(R) / 7 (W)
LINK DATA REGISTER	LD	0	9999	20000	R/W	10(R) / 7 (W)
FILE REGISTER	FL	0	9999	30000	R/W	10(R) / 7 (W)
SET VALUE AREA		0	9999	40000	R/W	10(R) / 7 (W)
ELAPSED VALUE AREA		0	9999	50000	R/W	10(R) / 7 (W)

Per i due esempi sotto riportati, si è selezionato il protocollo NAIS Matsushita per la porta EXP1.

L'istruzione illustrata di seguito scrive il contenuto delle 8 word da V10 a V17 del TD320 nel registro EXTERNAL OUTPUT del PLC NAIS da Y3 ad YA (Y10).

Numero bobina

EXP_ 1

Parametri

Azione e indirizzo slave

Srivi sullo SLAVE numero
1
Min 0
Max 255

Indirizzo Word\Bit

Bit numero
10003
Min 0
Max 65535

Area (Dest. per lettura Sor. per scrittura)

Area memoria V word
10

Numero Word\Bit letti\scritti consecutivi

N° word
8
Min 0
Max 16

L'istruzione illustrata di seguito invece legge dal registro DATA REGISTER del PLC NAIS le 10 word da DT0 a DT9 e le copia nelle aree da V0 a V9 del TD320.

Numero bobina	
EXP_	1
Parametri	
Azione e indirizzo slave	
Leggi dallo SLAVE numero	1 Min 0 Max 255
Indirizzo Word\Bit	
Word numero	10000 Min 0 Max 65535
Area (Dest. per lettura Sor. per scrittura)	
Area memoria V word	0
Numero Word\Bit letti\scritti consecutivi	
N° word	10 Min 0 Max 16

7 Programmazione Ladder del TD320

La programmazione della parte PLC del TD320 avviene tramite l'ambiente di sviluppo **PLProg 4.xx**, che mette a disposizione dell'utente le risorse necessarie per la creazione dello schema Ladder .

La compilazione ed il download, procedure discusse nel cap. 4, permettono di ottenere dal terminale TD320 il funzionamento desiderato.

Sono di seguito riportati tutti gli elementi disponibili (contatti e bobine) e le relative caratteristiche per la creazione dello schema.

7.1 Contatti ingressi digitali I

I contatti degli ingressi digitali I possono contenere lo stato degli ingressi letti tramite seriale da altri dispositivi, fino ad un massimo di 512.

Il contatto normalmente aperto è chiuso (ON) quando il bit vale 1 (ingresso attivo). Il contatto normalmente chiuso è aperto (ON) quando il bit vale 0 (ingresso non attivo).

7.2 Contatti uscite digitali Q

Il TD320 dispone di 512 uscite tipo "Q". Possono essere utilizzate per contenere lo stato di eventuali uscite di altri dispositivi, da comunicare tramite seriale.

Ciascuna uscita ha una bobina e relativo contatto logico N.A. ed N.C. All'eccitazione della bobina "Q" il relativo contatto logico si chiuderà (se normalmente aperto) o aprirà (se normalmente chiuso).

7.3 Relè bistabili B

Nel TD320 sono disponibili 128 relè bistabili. Ciascuno ha una bobina ed il relativo contatto logico N.A. e N.C. All'eccitazione della bobina "B" il relativo contatto logico cambierà di stato, se era

chiuso si apre, se era aperto si chiude. Il contatto normalmente aperto è chiuso (ON) quando il bit vale 1. Il contatto normalmente chiuso è aperto (ON) quando il bit vale 0. All'accensione del terminale il contatto N.A. risulta aperto.

7.4 Timer T

Il TD320 dispone di 128 timer da 16bit. Sono disponibili in tre modalità di funzionamento:

- **TON ritardo all'attivazione:** il tempo viene contato quando la bobina è attiva (ON). Il bit del timer (contatto T) viene attivato quando il valore corrente (T word) diventa maggiore o uguale al tempo preimpostato (preset, PT word). Quando la bobina non è attiva (OFF), il valore corrente del timer viene resettato. Il timer si arresta comunque al raggiungimento del valore massimo rappresentabile in 16 bit con segno (32767).
- **TOF ritardo alla disattivazione:** consente di ritardare la disattivazione di un'uscita per un dato periodo di tempo dopo che l'ingresso è stato disattivato. Quando la bobina viene attivata (ON), il bit del timer (contatto T) viene immediatamente attivato e il valore corrente (T word) viene impostato a 0. Alla disattivazione della bobina, il timer conta finché il tempo trascorso diventa pari a quello preimpostato (preset, PT word). Una volta raggiunto, il bit del timer si disattiva e il valore corrente smette di avanzare. Se l'ingresso resta disattivato per un tempo inferiore a quello preimpostato, il bit del timer resta attivo. Per iniziare il conteggio, l'operazione TOF deve rilevare una transizione da stato attivo a non attivo (ON → OFF).
- **TONR con memoria:** il tempo viene contato quando la bobina è attiva (ON). Il bit del timer (contatto T) viene attivato quando il valore corrente (T word) diventa maggiore o uguale al tempo preimpostato (preset, PT word). Quando la bobina è disattivata (OFF), il valore corrente del timer viene mantenuto. E' possibile così accumulare il tempo per più periodi di attivazione della bobina. Il valore corrente del timer può essere resettato con l'operazione MOV(Tx = #0). Il timer si arresta comunque al raggiungimento del valore massimo rappresentabile in 16 bit con segno (32767).

La base dei tempi può essere scelta tra 10ms, 100ms e 1s per ogni modalità di funzionamento.

Il valore corrente del timer è un multiplo della base di tempi selezionata. Ad esempio, un valore corrente di 50 in un timer con base dei tempi di 10ms corrisponde a 500 ms, con base dei tempi di 1s corrisponde a 50s.

Il preset del timer (PT) può essere una costante, oppure il contenuto di un'area VW, SMW, AI o TR.

7.5 Contatori C

Il TD320 dispone di 64 contatori da 16bit. Sono disponibili in due modalità di funzionamento:

- **MUP conta in avanti:** il bit di conteggio (contatto C) viene attivato quando il valore corrente (C word) è maggiore o uguale al valore preimpostato (PV). Il contatore conta in avanti ogni volta che l'ingresso di conteggio in avanti Cx(UP) è attivo e conta all'indietro ogni volta che l'ingresso di conteggio all'indietro Cx(DOWN) è attivo. Il contatore si azzerà quando si attiva l'ingresso di reset Cx(RESET) o quando viene eseguita l'operazione MOV(Cx = #0). Al raggiungimento del valore massimo (32767), il fronte di salita successivo dell'ingresso di conteggio in avanti lascerà invariato il valore corrente. Analogamente, al raggiungimento del valore minimo (-32768) il successivo fronte di salita dell'ingresso di conteggio all'indietro lascerà invariato il valore corrente. Per i contatori in avanti, il valore preimpostato (PV) viene confrontato con il valore corrente al termine di ogni ciclo del programma. Se è maggiore o uguale al valore preimpostato, il bit di conteggio si attiva (contatto C), altrimenti il si disattiva.
- **MDOWN conta all'indietro:** il bit di conteggio (contatto C) viene attivato quando il valore corrente diventa uguale a zero. Il contatore conta all'indietro da un valore predefinito (PV) sui fronti di salita dell'ingresso di conteggio all'indietro Cx(DOWN) e conta in avanti sui fronti di salita dell'ingresso di conteggio in avanti Cx(UP). Al raggiungimento del valore massimo (32767), il fronte di salita successivo dell'ingresso di conteggio in avanti

lascerà invariato il valore corrente. Il contatore resetta il bit di conteggio (contatto C) e carica preimpostato (PV) quando l'ingresso di caricamento Cx(RESET) diventa attivo. Il contatore in modalità conta indietro smette di contare quando raggiunge lo zero.

Il valore predefinito (PV) può essere una costante, oppure il contenuto di un'area VW, SMW, AI e TR.

7.6 Formule matematiche FM

La funzione formula matematica FM esegue operazioni matematiche (+, -, *, /, | OR logico, & AND logico, ^ XOR logico, << ROL shift a sinistra, >> ROR shift a destra) tra due operatori e salva il risultato in un'altra locazione di memoria. Gli operatori possono essere numerici (costanti), oppure fare riferimento alle aree di memoria disponibili.

7.7 Assegnazioni MOV

La funzione di assegnazione MOV assegna alla locazione di memoria specificata (area destinazione) un valore numerico (costante) o il contenuto di un'altra locazione (area sorgente). Un'istruzione del tipo MOV(A=B) copia il contenuto della locazione di memoria B nella locazione di memoria A.

7.8 Assegnazioni multiple BLKMOV

La funzione di assegnazione BLKMOV assegna al blocco di memoria destinazione, a partire dalla locazione di memoria specificata, un valore numerico o il valore assunto da un'altro blocco di locazioni di memoria.

Un'istruzione del tipo BLKMOV($A_i=B_i$, num. Dati 8) copia il contenuto della locazione di memoria B_i nella locazione di memoria A_i , il contenuto della locazione B_{i+1} nella locazione A_{i+1} ,....., il contenuto di B_{i+7} nella locazione A_{i+7} .

7.9 Assegnazioni indicizzate MOVIND

La funzione di assegnazione indicizzata MOVIND assegna alla locazione di memoria specificata da un'altra locazione di memoria, un valore numerico (costante) o il valore assunto da un'altra locazione di memoria selezionata nell'area specificata dal valore di un'altra locazione di memoria come indice.

Questo tipo di assegnazione permette di considerare le varie aree di memoria come dei vettori di N locazioni ciascuno, dove tramite il valore assunto da un'altra locazione come "indice", è possibile accedere al valore $n=0, n=1, \dots, n=N-1$ dell'area.

Un'istruzione del tipo MOVIND(A[B]=C[D]) copia il contenuto della locazione di memoria C[D] nella locazione A[B]. L'indice dell'area C è specificato da D, che può essere un'altra locazione di memoria, ed analogamente B è l'indice dell'area A.

7.10 Assegnazioni MOVTEXT

La funzione di assegnazione MOVTEXT salva, a partire dalla locazione di memoria specificata, i caratteri di una stringa passata come parametro alla funzione. La funzione permette i seguenti tipi di formattazione dei caratteri della stringa nell'area di memoria:

- UN_CARATTERE_PER_WORD in questo formato, ciascuna word dell'area di destinazione conterrà un solo carattere della stringa di partenza.
- DUE_CARATTERI_PER_WORD in questo formato, ciascuna word dell'area di destinazione conterrà due caratteri della stringa di partenza, iniziando dalla parte alta.

7.11 Contatti ingressi digitali immediati II

I contatti II consentono di leggere istantaneamente lo stato dell'ingresso digitale. Il contatto normalmente aperto è chiuso (ON) quando il bit vale 1 (ingresso attivo). Il contatto normalmente chiuso è aperto (ON) quando il bit vale 0 (ingresso non attivo).

7.12 Contatti IF

L'operazione confronto condizionale IF confronta i valori di due variabili di qualsiasi area di memoria. Si possono effettuare i seguenti tipi di confronto: = (uguale), >= (maggiore o uguale), <= (minore o uguale), > (maggiore), < (minore), <> (diverso). Il contatto è attivo quando il confronto è vero.

7.13 Funzioni SBIT e RBIT

La funzione SBIT porta a "1" un bit di un'area di memoria quando la bobina della funzione è allo stato attivo.

La funzione RBIT porta a "0" un bit di un'area di memoria quando la bobina della funzione è allo stato attivo.

L'indice del bit varia da 0 a 15 (l'area destinazione è sempre una word), dove per bit 0 si intende il bit meno significativo (LSB).

7.14 Contatti BIT

Questa operazione ricava il valore di un bit di un'area di memoria.

Il Contatto normalmente aperto è chiuso (ON) quando il bit vale 1.

Il Contatto normalmente chiuso è aperto (ON) quando il bit vale 0.

L'indice del bit va da 0 a 15 (l'area destinazione è sempre una word), dove per bit 0 si intende il bit meno significativo (LSB).

7.15 Funzioni RANGE

La funzione RANGE definisce il valore del limite minimo e massimo per gli ingressi analogici AI, per i trimmer TR, per le uscite analogiche AQ e per le uscite dei PID.

RANGE(AI1, Min 10, Max 200)

la funzione imposta per l'ingresso analogico AI1 il limite minimo a 10 e il limite massimo a 200. Se all'ingresso analogico AI1 corrispondesse un potenziometro (da un PLC tramite seriale), da utilizzare per impostare il preset (PT) di un timer con base tempi

100ms, si otterrà, a seconda della posizione del potenziometro, un tempo variabile da 1.0 a 20.0 secondi.

Se si impostano valori esterni all'intervallo specificato dalla funzione RANGE, l'uscita viene bloccata al valore minimo o massimo ammessi.

Per quanto riguarda le uscite PID, i valori minimo e massimo servono a calcolare il valore dell'uscita generata dall'algoritmo di regolazione. Prendiamo in considerazione l'esempio seguente:

RANGE(PID1, Min 100, Max 500)

la funzione imposta per l'uscita PID1 il limite minimo a 100 e il limite massimo a 500. Ciò significa che in corrispondenza di un'uscita dello 0%, corrisponderà un'uscita del PID pari al valore minimo impostato (100), e in corrispondenza del 100%, corrisponderà un'uscita pari al valore massimo (500).

7.16 Contatti NOT

Il contatto NOT modifica lo stato del flusso di corrente. Il flusso di corrente si arresta se raggiunge il contatto NOT e fornisce energia se non lo raggiunge.

L'operazione NOT inverte il valore logico ($0 \rightarrow 1$ e $1 \rightarrow 0$).

7.17 Contatti P e N

Il contatto transizione positiva P attiva il flusso di corrente per un ciclo di scansione ad ogni transizione da off a on. Il contatto transizione negativa N attiva il flusso di corrente per un ciclo di scansione ad ogni transizione da on a off.

Le istruzioni che seguono nello schema saranno eseguite quindi una sola volta (per un ciclo di scansione) per ogni transizione che attiva il contatto.

7.18 Funzioni SEND

La funzione SEND trasmette i dati tramite le porte seriali nella modalità free-port.

In questa modalità, attivabile tramite gli special marker SM39, SM40 e SM41, il protocollo che normalmente gestisce le porte

seriali viene disabilitato e il programma ladder prende il controllo delle porte e dei buffer di trasmissione e ricezione.

Dopo aver caricato il buffer con i dati da trasmettere, attivando la funzione SEND, che ha come parametri la porta seriale e il numero di caratteri da trasmettere, i dati verranno inviati nella linea seriale.

Durante la fase di trasmissione, i bit SM0.6, SM0.7 o SM0.8, relativi alla porta in trasmissione, verranno settati a 1, mentre alla fine della trasmissione saranno posti a 0. E' possibile controllare un'eventuale risposta di un dispositivo collegato tramite la gestione delle SM42, SM43 e SM44, che contengono il numero di caratteri ricevuti e salvati nel buffer di ricezione di ciascuna porta seriale. Qualsiasi scrittura su ciascuno di questi special marker equivale allo svuotamento del buffer dei dati in ricezione nella porta corrispondente.

Chiamate alla funzione SEND prima della fine della trasmissione precedente o con modalità free-port disabilitata verranno ignorate dal programma.

7.19 Funzioni TUNE POS e POS

La funzione "TunePOS" esegue una procedura di autotuning, indispensabile per ricavare i dati di tempo di reazione ed inerzia dell'asse su cui è richiesta una procedura di posizionamento.

La funzione "POS" esegue il posizionamento ON/OFF dell'asse. Le funzioni operano sull'area variabili VD (doppia word); l'indirizzo di inizio dell'area viene richiesto come parametro dalle funzioni "TunePOS" e "POS". La tabella seguente indica come sono organizzati i dati nell'area dalle due funzioni a partire dall'indirizzo della locazione specificata:

Indirizzo area VD	Contenuto
+0	Conteggi encoder
+2	Conteggi valore setpoint di posizionamento
+4	Conteggi scarto assoluto massimo del posizionamento

+6	Tempo impiegato per raggiungere la max velocità (in decimi di sec)
+8	Stato dell'uscita di posizionamento (0=fermo, 1=avanti, -1=indietro)
+10	Conteggi inerzia marcia avanti
+12	Conteggi inerzia marcia indietro
+14	Durata minimo impulso (risoluzione 0.2 mS)
+16	Conteggi spostamento dopo impulso di 100 mS
+18	Conteggi spostamento dopo impulso di 500 mS
+20	Conteggi spostamento dopo impulso di 1000 mS

Per un corretto funzionamento è necessario procedere come segue:

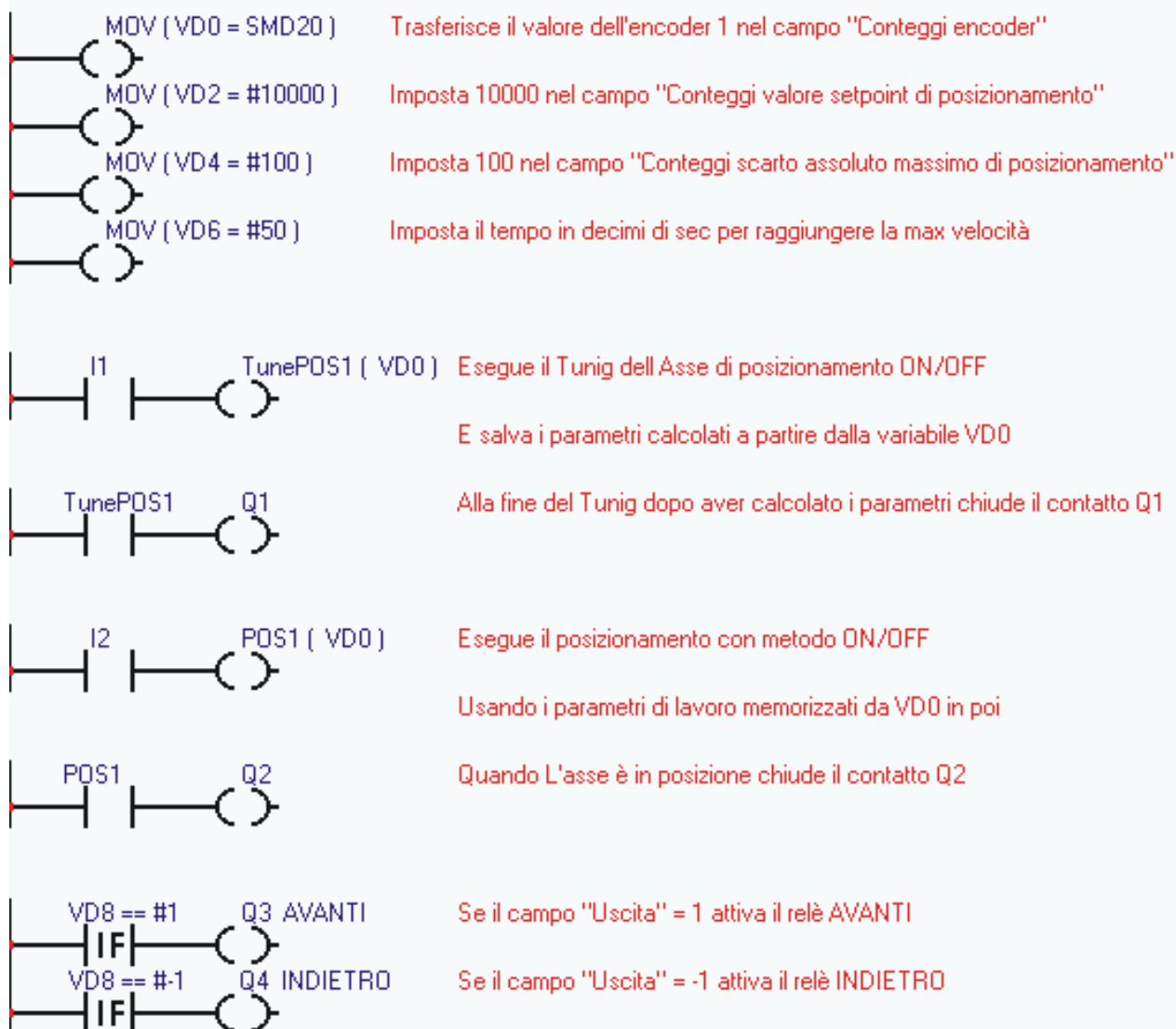
- Trasferire il conteggio dell'encoder connesso ad un dispositivo remoto (letto tramite seriale) nel campo "Conteggi encoder" (area di memoria di partenza).
- Impostare nel campo "Conteggi valore setpoint di posizionamento" il valore in conteggi a cui si vuole posizionare l'asse.
- Impostare nel campo "Conteggi scarto assoluto massimo del posizionamento" il valore in conteggi dello scarto massimo consentito al posizionamento.
- Impostare il tempo, in decimi di secondo, necessario perché l'asse raggiunga la massima velocità.
- Attivare la funzione "TunePOS" e attendere che il contatto TunePOS (normalmente aperto) si chiuda ad indicare la fine della procedura di autotuning dell'asse. A questo punto, i dati di inerzia e di tempo di reazione dell'asse vengono automaticamente memorizzati nell'area di memoria indicata, restando a disposizione per la funzione "POS".
- Disattivare la funzione "TunePOS".
- Attivare la funzione "POS". Quando l'asse si posizionerà sul set impostato (a meno dello scarto prefissato), il contatto POS (normalmente aperto) si chiuderà, ad indicare la fine del posizionamento.
- Attivare le uscite AVANTI e INDIETRO andando a leggere il valore del campo "Uscita" (VD+8). Se il valore di "Uscita" è "1",

bisogna attivare l'uscita AVANTI, se è "-1" bisogna attivare l'uscita INDIETRO, se è "0" non bisogna attivare nessuna uscita.

- Impostare a 0 il valore del campo "Uscita" quando viene tolto il consenso alla funzione "TunePOS" o "POS", per evitare che l'uscita rimanga impostata su avanti o indietro.

L'esempio seguente riporta il segmento di codice ladder che implementa il posizionamento dell'asse come spiegato nella procedura:

ESEMPIO DI UTILIZZO DELLE FUNZIONI TunePOS e POS



7.20 Funzioni COM ed EXP

Le funzioni di comunicazione COM ed EXP consentono di programmare le due porte seriali (COM1 ed EXP1) per la lettura/scrittura di dati dai dispositivi slave collegati, utilizzando il protocollo master selezionato nel progetto.

Tali funzioni sono attive solamente quando nel progetto è selezionato per la porta seriale corrispondente un protocollo di comunicazione di tipo master, cioè un protocollo che consenta al TD320 di prendere il controllo della linea andando a controllare il flusso dei dati verso i dispositivi slave.

Le due funzioni sono simili, cambia solamente la porta seriale a cui fanno riferimento. Tenere in considerazione che un'interfaccia RS485 permette di collegare sulla stessa linea più dispositivi, mentre l'interfaccia RS232 permette di collegare al TD320 un solo dispositivo.

Le istruzioni sono attive fino a che risulta attiva la bobina corrispondente, ma bisogna tenere presente che, a seconda del protocollo di comunicazione, i tempi di aggiornamento dei dati possono variare sensibilmente e che al momento dell'attivazione della bobina, i dati letti non sono disponibili istantaneamente, ma solo dopo un certo tempo legato ai ritardi di comunicazione.

L'istruzione COM ed EXP necessitano dei seguenti parametri:

- Indice (si possono impostare al massimo 256 interrogazioni seriali diverse su ogni porta)
- Tipo di operazione eseguita:
 - Lettura: il TD320 legge continuamente i dati dal/dai dispositivi slave e li memorizza in un'area di memoria interna.
 - Scrittura: il TD320 scrive continuamente i dati contenuti in un'area di memoria interna nel/nei dispositivi slave.
 - Lettura/Scrittura: il TD320 normalmente legge dei dati dal dispositivo slave e li memorizza in un'area di memoria interna; nel momento in cui tali dati interni al TD320 verranno modificati dal programma, la variazione verrà passata automaticamente al dispositivo slave tramite un'istruzione di scrittura (un solo dato alla volta).

- Numero dello slave (indirizzo di comunicazione del dispositivo slave)
- Tipo di dato (word o bit)
- Indirizzo ModBus relativo al dato (ai dati) da trasferire
- Area di memoria interna del PL250A dove leggere o scrivere i dati
- Numero di word (le istruzioni di lettura e scrittura, possono trasferire contemporaneamente fino a 16 bit/word consecutivi).

7.21 Funzioni StartPID, PID e SetOutPID

Le funzioni StartPID, PID e SetOutPID consentono la regolazione di una grandezza tramite algoritmo ad azione proporzionale, integrale e derivativa.

La funzione **StartPID** attiva la regolazione. La funzione può essere attivata una sola volta all'accensione oppure richiamata in un momento successivo permettendo la modifica “al volo” dei parametri di regolazione. L'azione integrale del PID viene azzerata solamente chiamando tale funzione e fissando il tempo integrale a 0. In caso contrario, anche in caso di spegnimento, il sistema inizierà a regolare mantenendo come punto di partenza la stessa percentuale di azione integrale, limitando quindi i tempi del transitorio.

Parametri della funzione **StartPID**:

- Banda proporzionale
- Tempo integrale
- Tempo derivativo
- Banda morta

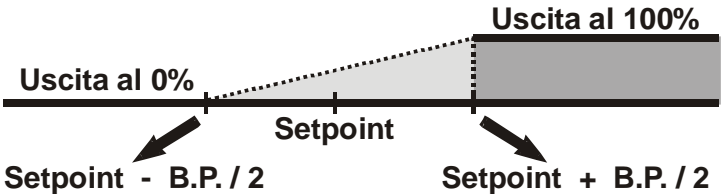
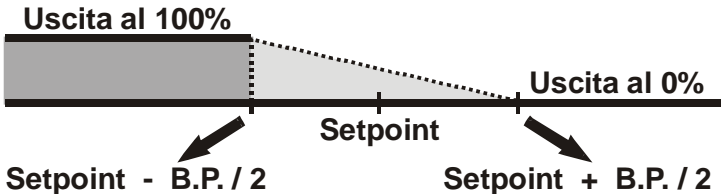
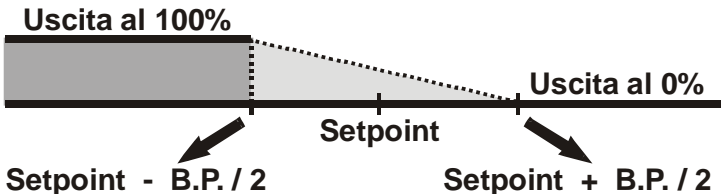
I parametri possono essere inseriti in formato numerico, oppure facendo riferimento a delle aree di memoria. Il tempo integrale è espresso nell'unità di tempo in cui viene richiamata la funzione PID (ad esempio, funzione PID richiamata ogni 1sec, tempo integrale espresso in secondi). Il tempo derivativo invece è espresso con una cifra decimale in più rispetto al tempo integrale. La banda proporzionale e la banda morta sono invece espresse in valore numerico pari al setpoint e al processo da regolare.

Parametri della funzione **PID**:

- Setpoint
- Processo
- Valore di uscita
- Tipo azione di regolazione

La funzione PID, dopo aver acquisito setpoint, processo, tipo di azione e tipo di uscita, imposterà nella variabile Valore di uscita il valore ottenuto dall'algoritmo di regolazione. Tale valore sarà ottenuto riscaldando il valore percentuale compreso tra 0 e 10000 (0.00% ÷ 100.00%) tra il valore minimo e massimo dell'uscita del PID impostati tramite la funzione RANGE.

La tabella seguente indica gli 8 tipi di regolazione e gli intervalli modulazione (l'effettivo valore entro l'intervallo è determinato anche dalle azioni integrali e derivative, in tabella si riportano le sole componenti proporzionali):

Tipo azione di regolazione	Intervalli di modulazione
Singola azione diretta, 0	 <p>Uscita al 0%</p> <p>Uscita al 100%</p> <p>Setpoint - B.P. / 2</p> <p>Setpoint</p> <p>Setpoint + B.P. / 2</p>
Singola azione diretta, 1	 <p>Uscita al 100%</p> <p>Uscita al 0%</p> <p>Setpoint - B.P. / 2</p> <p>Setpoint</p> <p>Setpoint + B.P. / 2</p>
Singola azione inversa, 0	 <p>Uscita al 100%</p> <p>Uscita al 0%</p> <p>Setpoint - B.P. / 2</p> <p>Setpoint</p> <p>Setpoint + B.P. / 2</p>

Singola azione inversa, 1	
Doppia azione diretta, 0	
Doppia azione diretta, 1	
Doppia azione inversa, 0	
Doppia azione inversa, 1	

La funzione PID, per un corretto funzionamento, deve essere richiamata ad intervalli il più possibile regolari, quindi si può utilizzare un timer, oppure per tempi più brevi e precisi, un interrupt interno.

La funzione **SetOutPID** va utilizzata per regolazioni che prevedono la doppia funzione automatico/manuale.

Serve ad evitare oscillazioni della grandezza di controllo nella commutazione da modo manuale ad automatico tramite l'algoritmo PID.

La funzione necessita dei seguenti parametri:

- Valore dell'uscita

Imposta il valore dell'uscita generata dal PID calcolando automaticamente le singole percentuali delle azioni proporzionale ed integrale. In questo modo, alla commutazione dal funzionamento manuale ad automatico, l'uscita del PID assumerà il valore impostato dal manuale e inizierà la regolazione.

La funzione deve quindi essere chiamata solo durante la fase di regolazione manuale, per mantenere così allineata l'uscita del PID con quella manuale. La funzione azzerava automaticamente l'azione derivativa. L'utilizzo di questa funzione con il processo fuori dalla banda proporzionale, fissa l'azione integrale a 0.

7.22 Funzioni GENSET

La funzione GENSET genera automaticamente un setpoint variabile crescente o decrescente, con la possibilità di impostare una rampa di accelerazione e una di decelerazione. La funzione GENSET opera su una serie di variabili in doppia word contigue, a partire dalla locazione indicata come parametro alla funzione.

La tabella seguente indica come sono organizzati i dati nell'area di memoria utilizzata dalla funzione a partire dall'indirizzo della locazione specificata:

Indirizzo area VD	Contenuto
+0	Stato della funzione GENSET 0 → Stop o fine spostamento 1 → Inizializzazione funzione 2 → Rampa di accelerazione 3 → Spostamento a velocità costante 4 → Rampa di decelerazione

+2	Setpoint iniziale / setpoint calcolato dalla funzione GENSET (conteggi)
+4	Setpoint finale (conteggi)
+6	Velocità di spostamento (conteggi*1000/unità tempo)
+8	Durata rampa di accelerazione (unità tempo)
+10	Durata rampa di decelerazione (unità tempo)
+12	Velocità istantanea del setpoint (conteggi*1000/unità tempo)

Per un corretto funzionamento è necessario procedere come segue:

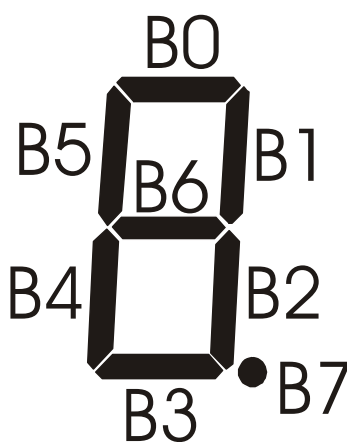
- Impostare nella locazione VD+2 il setpoint di partenza
- Impostare nella locazione VD+4 il setpoint finale
- Impostare nella locazione VD+6 la velocità massima di spostamento in conteggi*1000/unità tempo (in modo da avere 3 cifre decimali. Per es.: impostare 12345 corrisponde ad una velocità di 12.345 conteggi/unità tempo).
- Impostare nella locazione VD+8 la durata della rampa di accelerazione (espressa in unità tempo; se la durata della fase di accelerazione deve essere di 1 secondo, e la funzione GENSET viene chiamata da un'interrupt a 1 ms, impostare 1000)
- Impostare nella locazione VD+10 la durata della rampa di decelerazione.
- Scrivere "1" nella locazione VD (la locazione indicata come parametro della funzione). In questo modo si dà lo "start" alla funzione che automaticamente inizierà a scrivere nella locazione VD+2 il setpoint generato. La locazione VD verrà anch'essa aggiornata con lo stato attuale, mentre la locazione VD+12 verrà scritta con la velocità istantanea del setpoint espressa con tre cifre decimali.

Al termine dello spostamento, quando la locazione VD+2 raggiunge il valore del setpoint finale, automaticamente la funzione entrerà in una fase di standby, indicato dal valore "0" nella locazione VD. In questo modo, la funzione GENSET, può essere lasciata sempre abilitata, anche quando lo spostamento non è necessario.

7.23 Funzioni CONV

La funzione CONV converte il dato sorgente in uno dei formati disponibili:

- **TO_7SEG_SIGNED:** Converte il dato in ingresso (una word con segno -32768..32767) in un numero specificato di cifre già trasformate in codifica per display a 7 segmenti. Alla funzione verrà passato come parametro il numero di digit (cifre) da convertire, partendo dalla cifra meno significativa. Il dato codificato sarà salvato (una cifra per word) a partire dalla word di destinazione e poi in quelle successive a seconda di quante cifre sono richieste.
- **TO_7SEG_UNSIGNED:** E' analoga a quella sopra descritta, con la differenza che il dato di origine è inteso come word senza segno (0..65535). La codifica è composta da un bit a 1 se il segmento deve rimanere acceso, e da uno 0 se il segmento deve rimanere spento. L'associazione tra i bit e i segmenti del display è la seguente:



- **TO_ASCII_SIGNED:** Converte il dato in ingresso (una word con segno -32768..32767) in un numero specificato di cifre ASCII. Alla funzione verrà passato come parametro il numero di cifre da salvare. Il dato codificato sarà salvato (una cifra per word) a partire dalla word di destinazione e poi in quelle successive a seconda di quante cifre sono richieste.

- **TO_ASCII_UNSIGNED:** E' analoga a quella sopra descritta, con la differenza che il dato di origine è inteso come word senza segno (0..65535).

8 Note / Aggiornamenti

Handwriting practice lines consisting of 28 horizontal dashed lines.

Handwriting practice lines consisting of 28 horizontal dashed lines.

PIXSYS

Via Tagliamento, 18
30030 Mellaredo di Pianiga (VE)

www.pixsys.net

e-mail: sales@pixsys.net - support@pixsys.net

Software Rev. 4.64

2300.10.067-RevB 210907

