



Pacchetto software

## di programmazione per controllori SAIA® PCD

Programmazione, pianificazione del progetto e configurazione di sistemi  
su base PLC

Controls

saia-burgess

Smart solutions for comfort and safety

Manuale Utente

### Vantaggi del pacchetto software di programmazione PG5

- **Portabilità dei programmi:** i programmi PG5 possono essere eseguiti su tutte le piattaforme SAIA® PCD.
- **Organizzazione del programma in file** (contenenti diversi blocchi di programma) per semplificare l'utilizzo condiviso dei file di programma tra diversi controllori SAIA® PCD.
- **Accetta i programmi PG3 e PG4 esistenti.**
- **Ambienti di programmazione e di debugging** combinati in ciascun editor di programma.
- **Semplice programmazione delle pagine video dei terminali** con l'Editore HMI.
- **Potente set di istruzioni** supportato da macro e da direttive assembler.

### Caratteristiche del pacchetto software PG5

- **Symbol Manager:** gestisce tutti i simboli locali, globali e di rete o gruppi di simboli. L'allocazione automatica degli indirizzi permette di evitare il ricorso all'indirizzamento fisso.
- **Project Manager:** configura e gestisce anche complesse installazioni di reti di PCD, ne mostra la struttura e ne fornisce la documentazione.
- **Funzioni online** per la messa in servizio ed il rilevamento di errori via Ethernet-TCP/ IP, SAIA® S-Bus, modem, ecc.
- **Ambienti di programmazione integrati:**
  - FUPLA (programmazione con blocchi funzione)
  - S-Edit (IL, programmazione in lista istruzioni)
  - GRAFTEC (programmazione grafica sequenziale)
- **Editori di rete integrati** per SAIA® S-Bus, PROFIBUS DP e FMS, LONWORKS®.
- **Numerose librerie aggiuntive** permettono di ampliare la portata delle funzioni PG5

## Indice Generale

---

	<b>Prefazione</b>	
1	Guida Rapida PCD	1-3
2	Gestione Progetti	2-3
3	Risorse PCD	3-3
4	Programmazione FUPLA	4-3
5	Strutture dei Programmi	5-3
6	Programmazione Graftec	6-3
7	Programmazione in Lista Istruzioni (IL)	7-3
8	Strumenti Aggiuntivi	8-3
9	Reti Saia (Saia Networks: s-Net)	9-2
10	Profi-S-Bus	10-2
11	Ether-S-Bus	11-2
12	Profi-S-IO	12-2

## Prefazione

---

Questo documento rappresenta una introduzione ai controllori programmabili SAIA® PCD e non un manuale dettagliato per la messa in servizio dei medesimi. Il suo scopo è quello di concentrare l'attenzione sui punti essenziali per gli utenti che desiderano acquisire rapidamente una buona esperienza pratica. Per informazioni più dettagliate, si prega di fare riferimento ai file di help forniti con il pacchetto di programmazione stesso, oppure ai manuali dettagliati reperibili nel CD di documentazione.

Al fine di assicurare le condizioni ideali per il processo di formazione, si consiglia di procurarsi i programmi, la documentazione ed il materiale qui di seguito elencati:

- CD PG5 versione 1.4
- CD di documentazione 26/803
- 1 controllore PCD2.M480<sup>1</sup>
- 1 modulo PCD2.E110 con 8 ingressi digitali
- 1 modulo PCD2.A400 con 8 uscite digitali
- 1 cavo di programmazione PCD8.K111

Tutte le istruzioni necessarie per installare il pacchetto software PG5 1.4 sul computer sono incluse nel CD del software PG5 versione 1.4 (vedere sotto: CD:\PG5\InstallationGuide\_I.htm).

Si prega inoltre di notare che tutti i nomi in Inglese dei menu, delle istruzioni, delle opzioni e dei pulsanti previsti nel programma PG5 sono riportati in *corsivo* all'interno del presente manuale.

Vi auguriamo di completare con profitto la fase di formazione e di cimentarvi al più presto con progetti basati sui prodotti SAIA® PCD.

Il Vostro partner Saia-Burgess Controls AG.

---

<sup>1</sup> può essere utilizzato anche un altro PCD



## Indice

---

1	Guida Rapida PCD	1-3
1.1	Introduzione	1-3
1.2	Preparazione dell'hardware	1-4
1.2.1	Esempio: Illuminazione vano scale	1-4
1.2.2	Schema di collegamento del PCD2.M480	1-4
1.2.3	Preparazione del PCD2.M480	1-5
1.2.4	Collegamento	1-5
1.3	Realizzazione del programma	1-6
1.3.1	Installazione del software	1-6
1.3.2	Avvio di PG5	1-6
1.3.3	Apertura di un nuovo progetto	1-6
1.3.4	Configurazione	1-8
1.3.5	Aggiunta di un file di programma	1-10
1.3.6	Apertura di un file	1-11
1.3.7	Scrittura di un programma	1-11
1.4	Esecuzione e test del programma	1-15
1.4.1	“Costruzione” del programma (Build)	1-15
1.4.2	Trasferimento del programma al PCD (Download)	1-15
1.5	Ricerca e correzione degli errori (Debugging)	1-16
1.6	Correzione di un programma	1-17



# 1 Guida Rapida PCD

---

## 1.1 Introduzione

Come primo contatto con l'apparecchiatura PCD, proponiamo un approccio diretto: la realizzazione di una piccola applicazione legata alla vita di tutti i giorni. Anche se non si possiede alcuna esperienza sui prodotti SAIA, l'operazione potrà essere eseguita senza difficoltà. Tutto verrà accuratamente descritto in questo capitolo di guida rapida all'uso.

Questo esempio illustra come procedere alla messa in servizio di un PCD2.M480. Programmazione e test eseguiti con i tool di programmazione PG5.

I capitoli successivi sviluppano in dettaglio i contenuti del presente capitolo di guida rapida all'uso e forniscono maggiori informazioni quali la descrizione delle risorse disponibili, le strutture dei programmi e la programmazione in Lista Istruzioni.

## 1.2 Preparazione dell'hardware

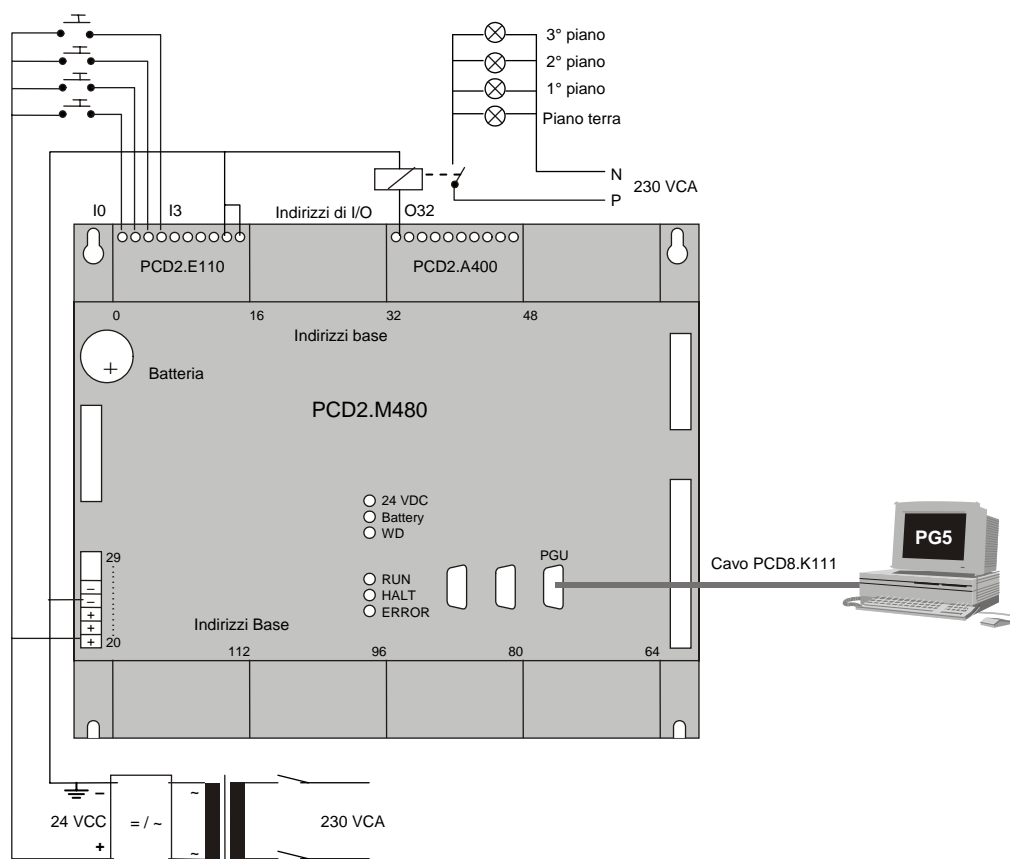
### 1.2.1 Esempio: Illuminazione vano scale

Viene qui illustrata la messa in servizio di un PCD usando come esempio un'applicazione dedicata all'illuminazione di un vano scale. L'edificio interessato è composto da un piano terra e da tre piani superiori. Per ogni piano è previsto un pulsante per l'accensione delle luci. Premendo brevemente questo pulsante, le luci di illuminazione del vano scale di tutti i 4 piani si accenderanno per un periodo di 5 minuti.

I pulsanti sono collegati ai seguenti 4 ingressi del PCD: I0, I1, I2 e I3.

L'accensione/spegnimento delle 4 luci avviene per mezzo di un relè. Questo relè è pilotato da un'unica uscita (O32) del PCD.

### 1.2.2 Schema di collegamento del PCD2.M480



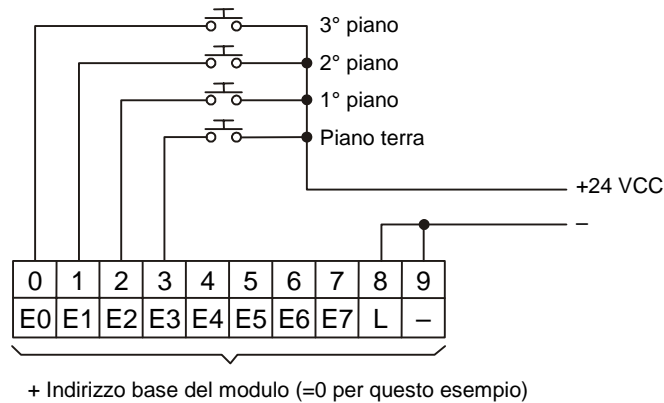


### 1.2.3 Preparazione del PCD2.M480

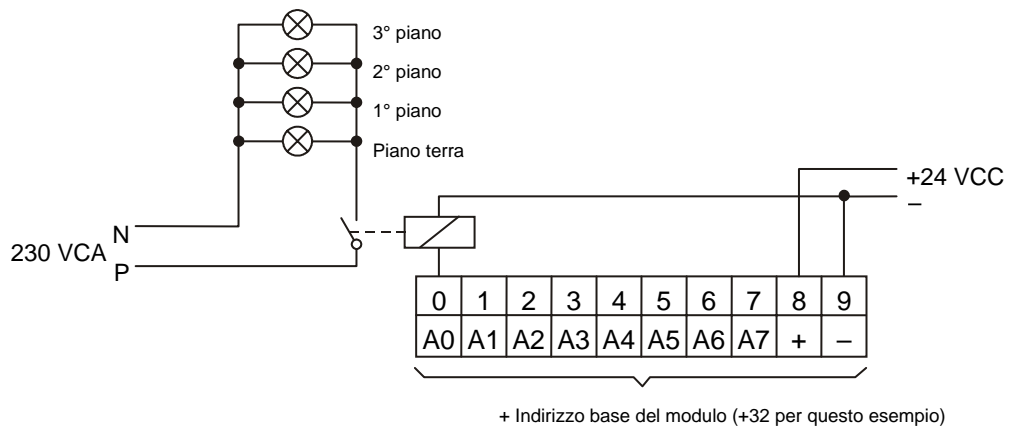
1. Inserire la pila al litio da 3,0 V inclusa nella fornitura.
2. Innestare un modulo PCD2.E110 all'interno dell'alloggiamento (socket) 1 (indirizzi da 0 a 15).
3. Spingere il modulo verso il centro del dispositivo, finché la parte terminale non si arresta e si blocca in posizione. Questo modulo fornisce 8 ingressi digitali a 24 VCC aventi indirizzi da I0 a I7. Verranno usati solo gli indirizzi da I0 a I4.
4. Innestare un modulo PCD2.A400 all'interno dell'alloggiamento (socket) 3 (indirizzi da 32 a 47) operando come precedentemente descritto. Questo modulo fornisce 8 uscite digitali (da O32 a O39) a 24 VCC / 0,5 A. Verrà usata solo l'uscita O32.

### 1.2.4 Collegamento

1. Collegare l'alimentazione 24 VCC ai morsetti a vite 20 (+) e 23 (-). Sono ammesse le seguenti tensioni di alimentazione: 24 VCC  $\pm 20\%$  filtrata o 19 VCA  $\pm 15\%$  raddrizzata
2. I quattro ingressi utilizzati sono collegati secondo quanto riportato nella descrizione hardware del modulo PCD2.E110. Collegare i 4 pulsanti ai morsetti da 0 a 3. I morsetti 8 e 9 sono collegati al polo negativo dell'alimentazione.



3. Come riportato nella descrizione hardware del modulo PCD2.A400, collegare la bobina del relè al morsetto 0, il polo positivo dell'alimentazione 24 VCC al morsetto 8 ed il polo negativo al morsetto 9.



4. Collegare l'interfaccia RS 232 del PC (porta COM) al connettore PGU del PCD. Per realizzare questo collegamento, utilizzare il cavo PCD8.K111.

Per maggiori dettagli circa l'installazione ed il collegamento della parte hardware del sistema, consultare il manuale hardware del PCD utilizzato.

## 1.3 Realizzazione del programma

### 1.3.1 Installazione del software

Installare sul PC il pacchetto di programmazione PG5 per SAIA@PCD (se non è ancora stato installato), seguendo le istruzioni contenute nel relativo CD (cd:\PG5\InstallationGuide\_I.htm).

### 1.3.2 Avvio di PG5

Avviare il Project Manager PG5 (*Gestore Progetti*):

**Start (Avvio) --> Programs (Programmi) --> SAIA PG5 V1.4 --> Project Manager (Gestore Progetti)**

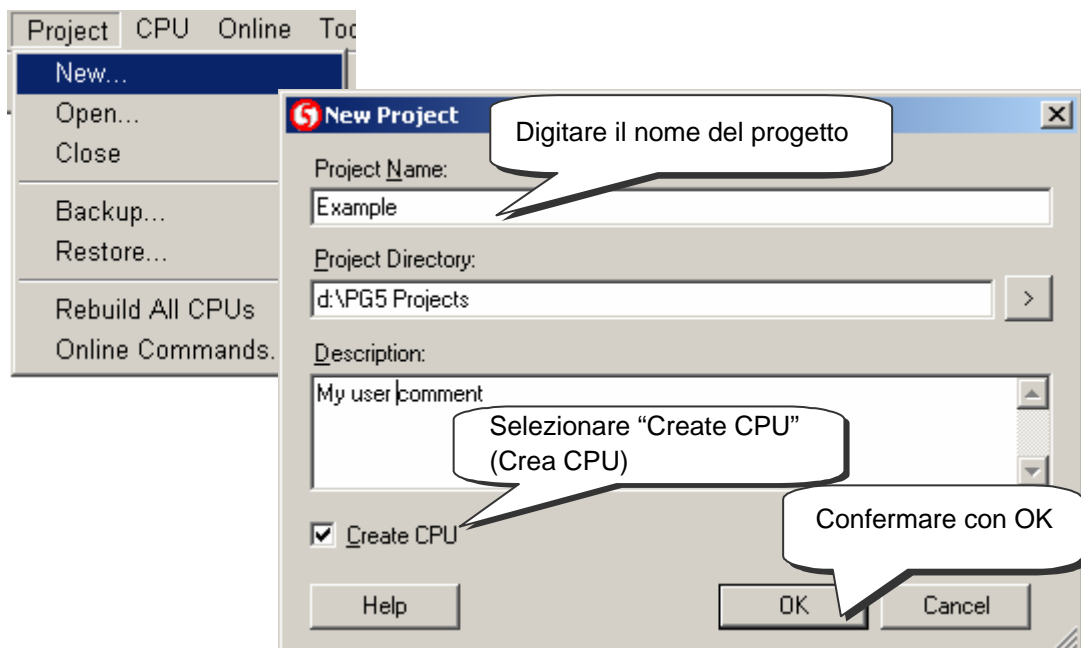


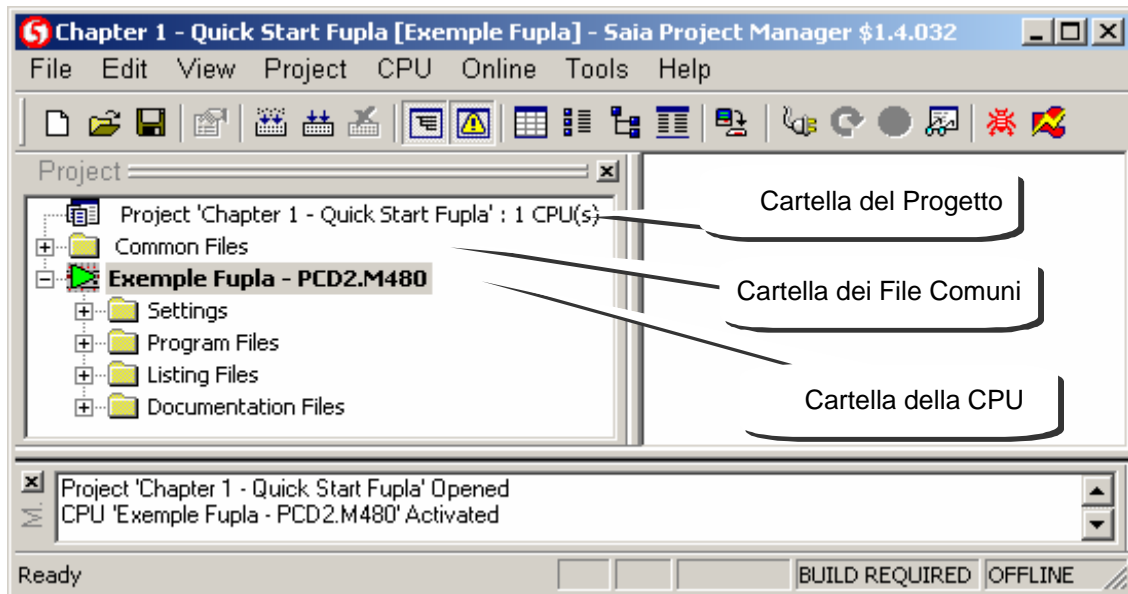
### 1.3.3 Apertura di un nuovo progetto

Prima di iniziare a scrivere un nuovo programma, è necessario creare un nuovo progetto o aprirne uno esistente che contenga le definizioni necessarie, alcuni parametri di configurazione ed i file necessari al programma utente.

Se il progetto non esiste ancora, selezionare le voci di menu *Project, New...*, (*Progetto, Nuovo...*), digitare il nome del nuovo progetto all'interno del campo *Project Name (Nome Progetto)*, selezionare la casella *Create CPU (Crea CPU)* e confermare con il pulsante *OK*.

#### Esecuzione di un nuovo progetto





Sullo schermo è già presente la finestra *SAIA Project Manager (Gestore Progetto SAIA)*. La finestra *Project (Progetto)* riporta la struttura del nuovo progetto (se questa finestra non è visualizzata, selezionare la voce di menu *View, Project Tree (Visualizza, Struttura Progetto)*).

Le cartelle riportate all'interno della finestra *Project (Progetto)* contengono le informazioni relative al progetto stesso, organizzate in base ai seguenti criteri:

- L'identificatore della cartella principale riporta sia il nome del progetto che il numero di CPU utilizzate nel progetto stesso.
- I moduli condivisi da più CPU possono essere memorizzati nella cartella *Common Files (File Comuni)*.
- Di seguito sono riportate le cartelle CPU (ciascuna CPU corrisponde ad un PCD).

Ogni cartella CPU contiene le seguenti sotto-cartelle:

- *Settings (Impostazioni)* Contiene la configurazione del pacchetto di programmazione e del PCD.
- *Program Files (File di Programma)* Contiene i file relativi al programma.
- *Listing Files (File Listato)* Contiene i file generati durante la costruzione del programma (*Build*). Questi file sono di scarsa importanza per gli utenti inesperti.

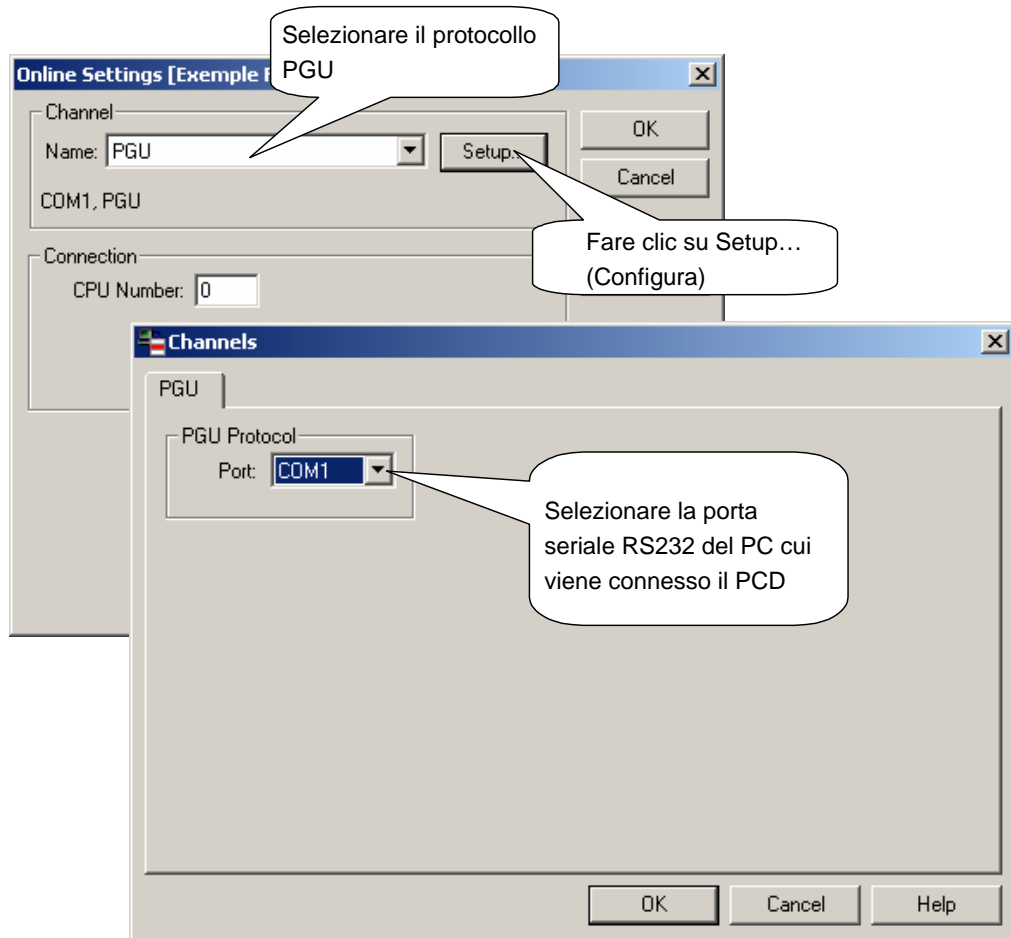
Apertura di un progetto esistente

Un progetto pre-esistente può essere aperto usando la voce di menu *Project, Open... (Progetto, Apri...)*. Questo comando provvede a ricercare tutti i file di progetto (.5pj) all'interno della directory progetto, visualizzandoli sotto forma di elenco. Fare doppio clic sul nome del progetto desiderato all'interno dell'elenco oppure selezionarlo e premere il pulsante *Open (Apri)*. In alternativa, è possibile premere il pulsante *Browse (Scorri)* e ricercare il file Progetto o il file CPU direttamente.

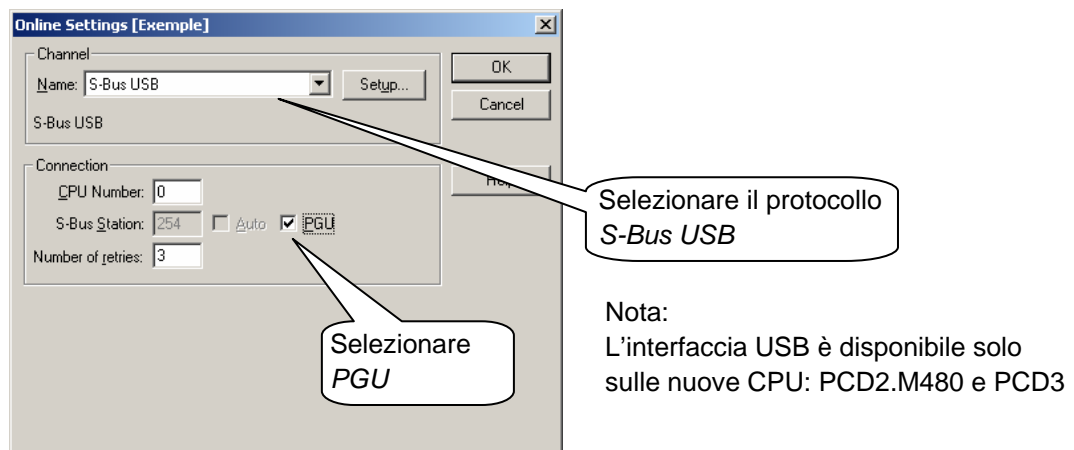
### 1.3.4 Configurazione

Prima di poter operare con una CPU all'interno del progetto, è necessario definire i parametri di configurazione, per consentire al pacchetto di programmazione ed al programma utente di interagire con il PCD.

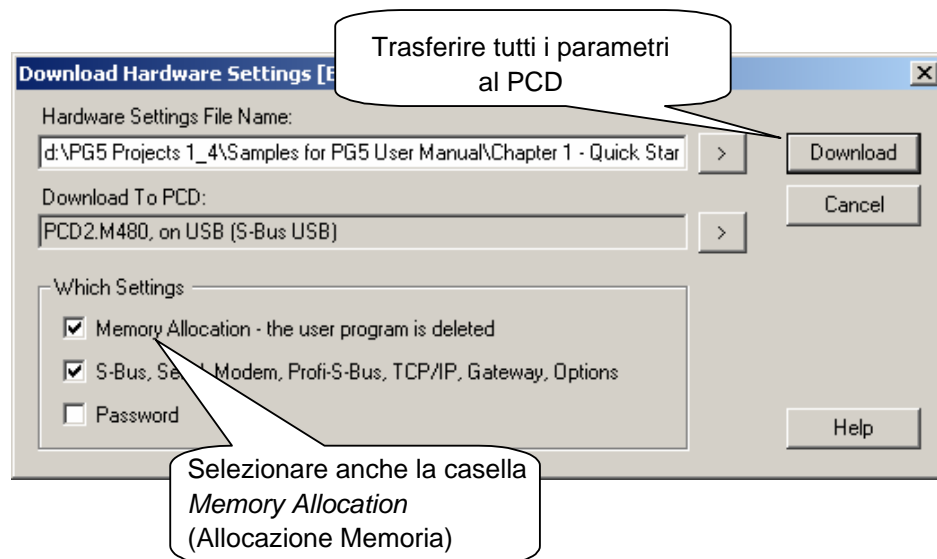
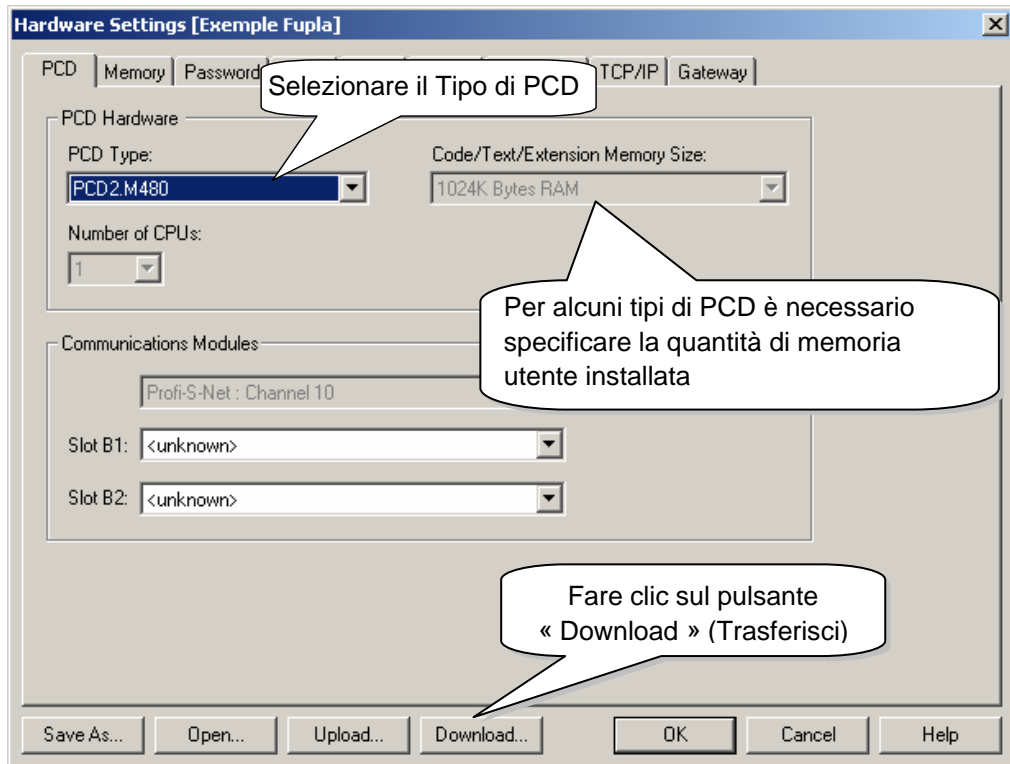
La finestra *Online Settings (Impostazioni Collegamento in Linea)* è possibile definire i parametri relativi alla comunicazione tra PC e PCD. Sono disponibili diverse possibilità. Per questa esercitazione, selezionare il protocollo di default (PGU) e successivamente il numero della porta seriale del PC (COM1).



#### Protocollo (Canale) S-Bus USB



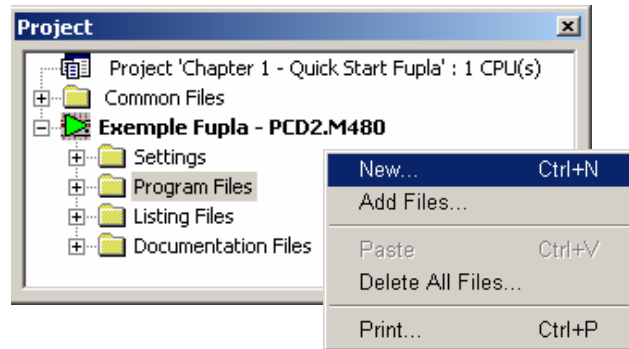
Il PCD viene configurato per mezzo della finestra *Hardware settings (Impostazioni Hardware)*. Per questa esercitazione, le funzioni *Memory, S-Bus, Gateway, Modem e Password* non sono necessarie. Tuttavia, è importante selezionare correttamente il tipo di PCD e la dimensione della memoria installata. Il PCD2.M480 viene sempre fornito con una quantità standard di 1024 Kbyte di memoria RAM.



### 1.3.5 Aggiunta di un file di programma

I programmi utente PCD sono memorizzati in uno o più file. Esistono vari modi per aggiungere un file di programma:

All'interno della finestra *Project (Progetto)*, selezionare la cartella *Program Files (File di Programma)*, fare clic con il pulsante destro del mouse per visualizzare il menu contestuale quindi selezionare la voce *New... (Nuovo)*.



Metodi alternativi:

Fare clic sul pulsante *New File (Nuovo File)* all'interno della barra degli strumenti oppure selezionare la voce di menu *File, New... (File, Nuovo...)*.

All'interno della finestra *New File (Nuovo File)* è possibile definire il nome ed il tipo del modulo: due elementi di informazione molto importanti.

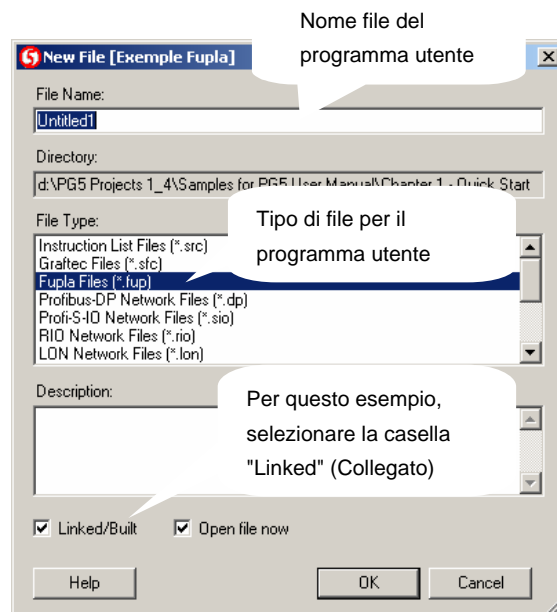
Per scrivere i programmi utente PCD sono disponibili vari editor.

L'utente può scegliere l'editore che meglio si adatta al programma utente da realizzare.

Per questa esercitazione, l'editore ottimale è:

*Fupla File (\*.fup)*.

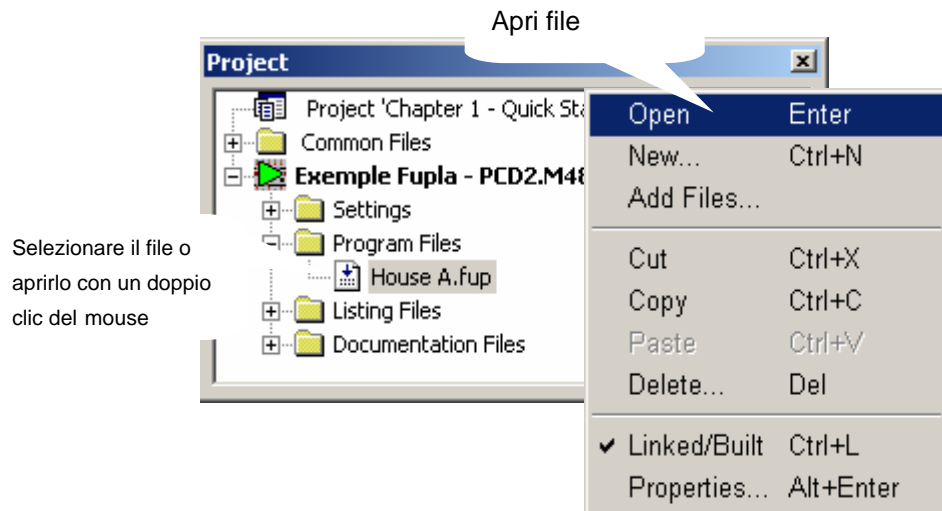
Fupla è un linguaggio di programmazione per scopi generali.



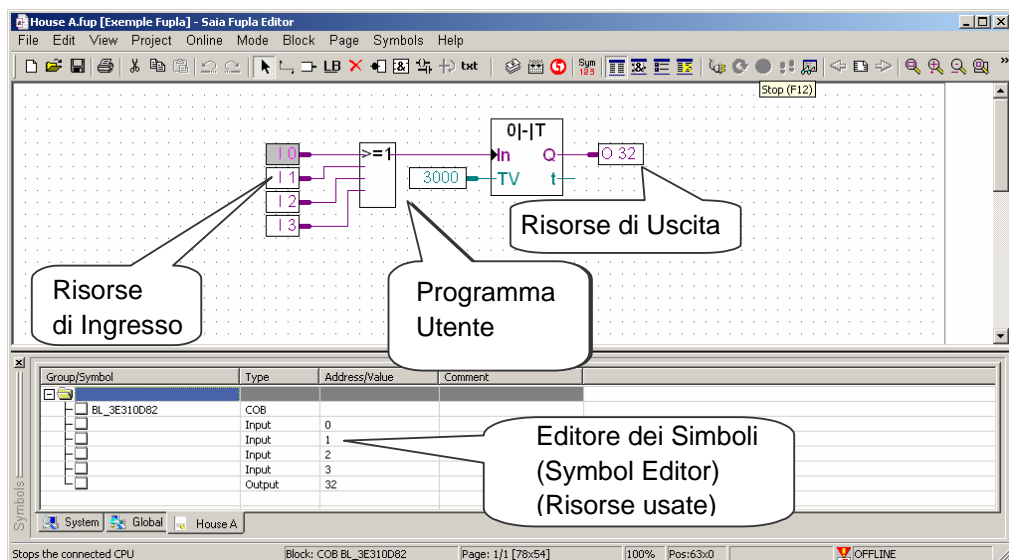
### 1.3.6 Apertura di un file

Se la cartella contiene già un file di programma, questo può essere aperto come di seguito indicato:

All'interno della finestra *Project (Progetto)*, aprire la cartella *Program Files (File di Programma)* e fare doppio clic sul file desiderato. In alternativa, fare clic con il pulsante destro sul nome del file per aprire il menu contestuale, quindi selezionare la voce *Open (Apri)*.



### 1.3.7 Scrittura di un programma



## Definizione delle risorse

Le risorse sono i dati necessari al programma utente PCD, come ad esempio i pulsanti per l'illuminazione del vano scale. Le risorse (simboli) sono definibili all'interno dei connettori presenti sulla pagina Fupla. Le risorse di 'Ingresso' (lettura) sono poste sulla sinistra mentre quelle di 'Uscita' (scrittura) sulla destra. Questo esempio di applicazione per l'illuminazione di un vano scale prevede 4 ingressi (I 0, I 1, I 2 e I 3) corrispondenti ai pulsanti di accensione luci ed un'uscita (O 32) che provvede a pilotare il relè di accensione/spegnimento luci. Il periodo di 5 minuti richiesto, durante il quale le luci del vano scale dovranno rimanere accese, deve essere digitato all'interno del connettore di ingresso, sotto forma di multiplo di decimi di secondo. Il valore da assegnare a questa costante è quindi 3000 (5 min. x 60 sec. x 10 = 3000).



### Inserisci Connettori

Per aggiungere un connettore ed il relativo simbolo (risorsa) ad una pagina Fupla, selezionare il pulsante *Place Connector (Inserisci Connettore)* della barra degli strumenti quindi posizionare il cursore del mouse all'interno della pagina Fupla. Per inserire un connettore di ingresso (lettura), è sufficiente fare poi clic con il tasto sinistro del mouse. Per inserire invece un connettore di uscita (scrittura), tenere premuto il tasto *Shift (Maiusc)* quindi fare clic con il tasto sinistro del mouse. Il connettore appena inserito risulterà immediatamente predisposto alla definizione del relativo simbolo; al suo interno verrà infatti visualizzato il cursore di inserimento. Se non si desidera definire immediatamente il simbolo, premere il tasto *ESC* e procedere all'inserimento del connettore successivo.

Per definire o modificare il simbolo (risorsa) di un connettore già presente sulla pagina Fupla, è sufficiente selezionare il connettore interessato facendo doppio clic su di esso. All'interno del connettore verrà visualizzato il cursore di inserimento. A questo punto è possibile inserire l'indirizzo di ingresso I 0 ... I 3 o di uscita O 32 oppure la costante. Accertarsi di lasciare sempre uno spazio tra la lettera I e l'indirizzo di ingresso. Lo stesso vale per la definizione della risorsa di uscita.



### Mostra/ Nascondi Editore dei Simboli

Per definire le risorse di ingresso, selezionare con il mouse 4 celle consecutive all'interno della colonna di sinistra e digitare gli indirizzi da I 0 a I 3. La costante di tempo 3000 (colonna di sinistra) e la risorsa di uscita O 32 (colonna di destra) sono definibili nello stesso modo. Notare che il tipo di indirizzo (I o O) ed il corrispondente valore (da 0 a 3 e 32) devono essere separati da un carattere "spazio". Le risorse verranno immediatamente visualizzate all'interno della finestra *Symbols (Simboli)* dell'editore dei simboli. Se quest'ultimo non è visibile, può essere visualizzato selezionando la voce di menu *View, Symbols (Visualizza, Simboli)* oppure premendo il pulsante *Show/Hide Symbol Editor (Mostra/Nascondi Editore dei Simboli)* all'interno della barra degli strumenti:

#### Nota:

Per default, su ogni nuova pagina creata viene predisposta una serie di connettori di ingresso (a sinistra) e di uscita (a destra). Se non si desidera che le nuove pagine contengano le suddette serie di connettori ma si preferisce posizionarli manualmente in base alle specifiche esigenze, disattivare la relativa opzione selezionando la voce di menu: *View, Options..., Layout, New pages with side connectors (Visualizza, Opzioni..., Imposta pagina, Nuove pagine con connettori laterali)*.

Per rimuovere i connettori vuoti presenti sulla sinistra o sulla destra della pagina, selezionare la voce di menu: *Page, Remove Empty connectors (Pagina, Rimuovi Connettori Vuoti)*.

Per reinserire nuovamente le serie di connettori vuoti all'interno di una pagina vuota, selezionare la voce di menu: *Page, Add Empty Side Connectors (Pagina, Inserisci Connettori Laterali Vuoti)*.



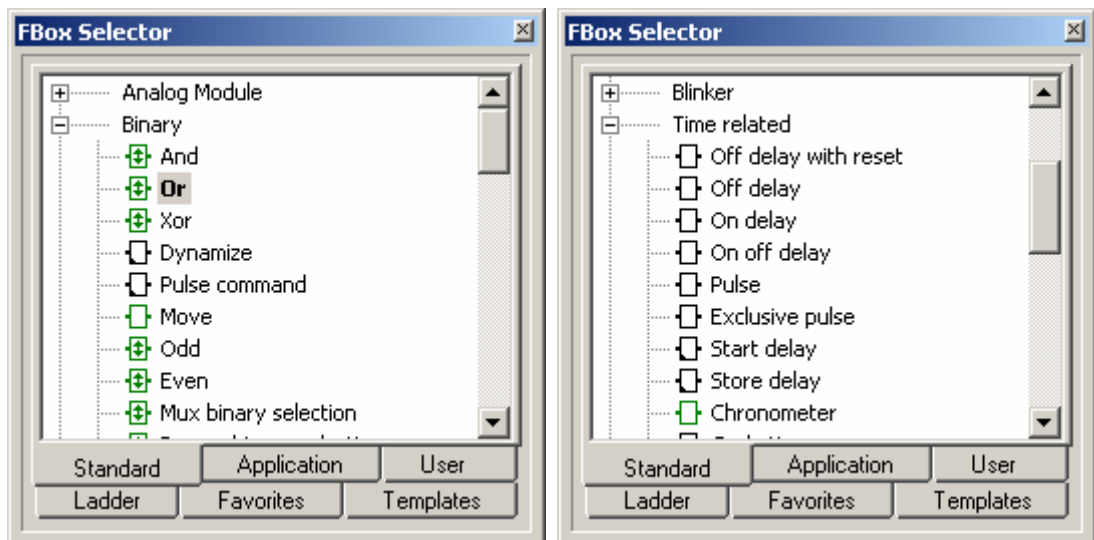
## Definizione delle funzioni di programma

Le funzioni di programma vengono inserite nell'area posta tra i connettori di 'ingresso' (lettura) e di 'uscita' (scrittura). Questa operazione viene eseguita posizionando opportunamente i simboli grafici relativi ai blocchi funzione (FBox) utilizzati per creare i programmi utente.

I blocchi funzione sono selezionabili all'interno della finestra *FBox Selector* (*Selezione FBox*).



Aggiungi Fbox



La prima funzione richiesta dall'esercitazione in oggetto serve per accendere le luci in risposta ad un breve impulso inviato da uno dei pulsanti di illuminazione vano scale. Questa è quindi una funzione OR, selezionabile all'interno della famiglia *Binary* (*Binarie*) della libreria *Standard*.

La seconda funzione (*Off delay – Spegnimento Ritardato*) definisce l'intervallo di 5 minuti durante il quale le luci restano accese. Questa funzione è selezionabile all'interno della famiglia *Time Related* (*Temporizzatori*) della libreria *Standard*. Ulteriori informazioni sugli FBox selezionati possono essere visualizzate facendo clic con il pulsante destro del mouse sull'identificatore della funzione riportato all'interno della finestra *FBox Selector* (*Selezione FBox*) e selezionando la voce *FBox Info* (*Informazioni su FBox*) all'interno del menu contestuale.

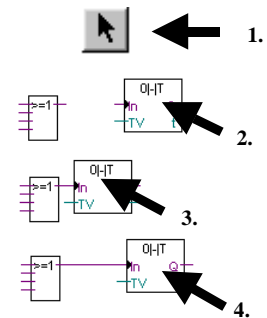
Dopo aver selezionato un blocco funzione all'interno della finestra di *Selezione FBox*, è possibile usare il pulsante sinistro del mouse per posizionare il blocco all'interno della finestra dell'editore, tra le due colonne di definizione delle risorse.

Per alcuni blocchi funzione, quali l'*OR logico*, è possibile selezionare il numero di ingressi collegati. Questa operazione può essere eseguita trascinando verticalmente il mouse e premendo il pulsante sinistro al raggiungimento del corretto numero di ingressi.

## Funzione Collegamento

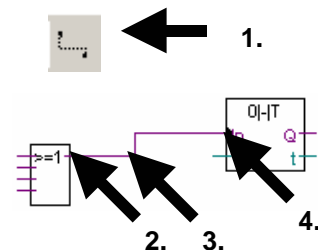
Usare questo metodo quando i punti da collegare sono allineati orizzontalmente.

1. Premere il pulsante *Select Mode (Modo Selezione)*
2. Posizionare il cursore del mouse sul simbolo dell'FBox quindi premere il pulsante sinistro del mouse.
3. Mantenere premuto il pulsante e trascinare orizzontalmente l'FBox finché non viene realizzato il collegamento. Non rilasciare il pulsante del mouse.
4. Riportare, trascinandolo, l'FBox nella posizione originale, quindi rilasciare il pulsante del mouse.



Usare questo metodo per le altre connessioni

1. Premere il pulsante *Auto Lines Mode (Modo Linee Automatico)*
2. Fare clic con il pulsante sinistro del mouse sul punto iniziale del collegamento quindi rilasciare il pulsante. Spostare il cursore del mouse verso destra, fino al punto desiderato, quindi ripremere il pulsante sinistro del mouse.
3. Spostare il mouse verticalmente e fare clic nuovamente con il pulsante sinistro del mouse.
4. Portare il cursore del mouse sul connettore dell'FBox e premere nuovamente il tasto sinistro del mouse per completare il collegamento.
5. Se necessario, l'operazione di collegamento può essere interrotta premendo il pulsante destro del mouse.



## Cancellare una Linea di collegamento, un Fbox, un Simbolo o un Connettore

Premere il pulsante *Delete Mode (Modo Cancellazione)* all'interno della barra degli strumenti e fare clic sulla Linea di collegamento, sull'Fbox, sul Simbolo o sul Connettore da cancellare.



## 1.4 Esecuzione e test del programma

### 1.4.1 “Costruzione” del programma (Build)



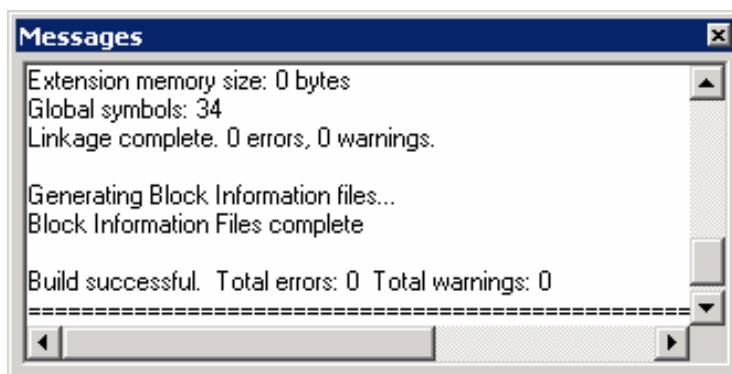
*Costruisci  
Tutti i File*

Prima che il programma possa essere eseguito dal PCD, è necessario procedere alla sua “costruzione” (compilazione, assemblaggio e link (*collegamento*)) usando la voce di menu *CPU, Rebuild All Files (CPU, Costruisci Tutti i File)* del *Project Manager (Gestore Progetti)* oppure il pulsante *Rebuild All Files (Costruisci Tutti i File)* nella barra degli strumenti dell'Editore Fupla o del *Project Manager*.

I risultati di questa “costruzione” sono riportati all'interno della finestra *Messages (Messaggi) (Compiling, Assembling, Linking etc. - compilazione, assemblaggio, collegamento, ecc.)*. Se il programma è stato definito correttamente, al termine della funzione di “costruzione” viene visualizzato il messaggio:

*Build successful. Total errors: 0 Total warnings: 0  
(Costruzione eseguita con successo. Errori Totali: 0 Avvertenze Totali: 0)*

Gli errori sono indicati da un messaggio di errore di colore rosso. La maggior parte degli errori può essere individuata all'interno del programma utente facendo doppio clic sul relativo messaggio di errore.



### 1.4.2 Trasferimento del programma al PCD (Download)

A questo punto il programma utente è pronto. Tutto ciò che resta da fare è provvedere al suo trasferimento dal PC al PCD. Questa operazione può essere eseguita servendosi del pulsante *Download Program (Trasferimento Programma)* nella barra degli strumenti del *Project Manager* oppure selezionando la voce di menu *Online, Download Program (In Linea, Trasferimento Programma)*.



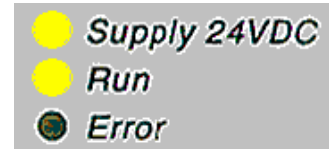
*Trasferimento  
Programma*

Se si verifica un qualsiasi problema di comunicazione, verificare le impostazioni di configurazione (*Settings Online (Impostazioni – In linea)* e *Settings Hardware (Impostazioni – Hardware)*) ed il collegamento del cavo (PCD8.K111 o USB) tra PC e PCD.

## 1.5 Ricerca e correzione degli errori (Debugging)

La prima versione di un programma non sempre risulta perfetta. E' pertanto necessario eseguire sempre un test accurato. Questo test del programma può essere eseguito utilizzando lo stesso editore usato per scrivere il programma.

1. Premere il pulsante *Go On /Offline (Passa In Linea/Fuori Linea)*
2. Avviare il programma con il pulsante *Run (Esegui)*



Contemporaneamente osservare il LED *RUN* presente sul PCD.

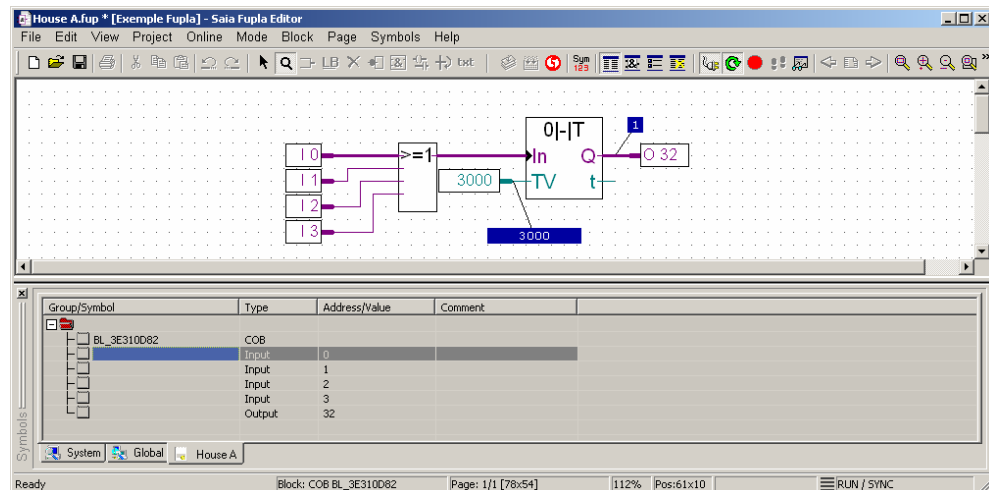
Alla pressione del pulsante *Run (Esegui)*, il LED *RUN* sul PCD deve accendersi, dal momento che il PCD sta ora eseguendo il programma utente.



Alla pressione del pulsante *Stop*, il LED *RUN* sul PCD deve invece spegnersi, dal momento che il PCD ha interrotto l'esecuzione del programma.

Quando l'editore è in modalità *Online (In Linea)* ed il PCD è in modo *RUN*, è possibile visualizzare lo stato di ogni singola risorsa:

- Lo stato logico relativo a dati binari viene indicato con una linea spessa o sottile (linea spessa = 1 e linea sottile = 0)
- I valori relativi ad altri tipi di dati sono visualizzabili facendo clic con il pulsante sinistro del mouse sul collegamento per visualizzare una finestra *Probe (Sonda)*: utilizzando il mouse, selezionare quindi il pulsante *Place Probe (Posiziona Sonda)* e poi il collegamento interessato.



## 1.6 Correzione di un programma

Per modificare un programma, operare come di seguito indicato:

1. Passare in modalità Fuori Linea (usando il pulsante *Go On /Offline (Passa In Linea/Fuori Linea)*).
2. Modificare il programma
3. Ri-costruire il programma (usando il pulsante *Build (Costruisci)* nella barra degli strumenti).
4. Trasferire il programma corretto al PCD (usando il pulsante *Download Program (Trasferimento Programma)* nella barra degli strumenti del Project Manager)





## Indice

---

2	Gestione Progetti	2-3
2.1	Introduzione	2-3
2.2	Organizzazione del progetto	2-4
2.2.1	Esempio di progetto	2-4
2.2.2	Salvataggio del progetto su PC	2-6
2.2.3	Compressione di un progetto o di una CPU	2-6
2.2.4	Apertura di un progetto	2-7
2.2.5	Creazione di un nuovo progetto	2-7
2.3	Finestra <i>Project (Progetto)</i>	2-8
2.3.1	Cartella Progetto	2-8
2.3.2	Cartella File Comuni	2-9
2.3.3	Cartella CPU	2-9
2.3.4	Impostazioni In Linea (Online)	2-10
2.3.5	Collegamento PC-PCD	2-10
2.3.6	Impostazioni Hardware	2-11
2.3.7	Impostazioni Software	2-16
2.3.8	Cartella File di Programma (Program Files)	2-18
2.3.9	Tipi di file	2-19
2.3.10	File Collegati ( <i>Linkati</i> )	2-20
2.3.11	File Comuni ( <i>Common files</i> )	2-20
2.4	Costruzione del programma	2-21
2.4.1	<i>Ricostruisci Tutto o Costruisci (Rebuild All e Build)</i>	2-22
2.4.2	<i>Opzioni della funzione di "Costruzione" (Build)</i>	2-22
2.5	Finestra Messaggi ( <i>Messages</i> )	2-23
2.6	Trasferimento del Programma nel PCD ( <i>Download Program</i> )	2-24
2.6.1	Opzioni di Trasferimento ( <i>Download options</i> )	2-25
2.6.2	Caricamento programma su memoria di backup (scheda Flash)	2-26
2.6.3	Backup della memoria e trasferimento del programma applicativo	2-26
2.7	Finestra di Visualizzazione ( <i>View</i> )	2-27
2.7.1	Struttura dei blocchi organizzativi	2-27
2.7.2	Elenco dei blocchi organizzativi	2-27
2.7.3	Elenco dei Simboli	2-28
2.7.4	Riferimenti Incrociati ( <i>Cross-Reference</i> )	2-28
2.8	Backup del Programma	2-29
2.9	File a trasferimento autonomo	2-29
2.9.1	Preparazione di un file <i>'sd5'</i>	2-30
2.9.2	Trasferimento di in file <i>'sd5'</i>	2-31





## 2 Gestione Progetti

---

### 2.1 Introduzione

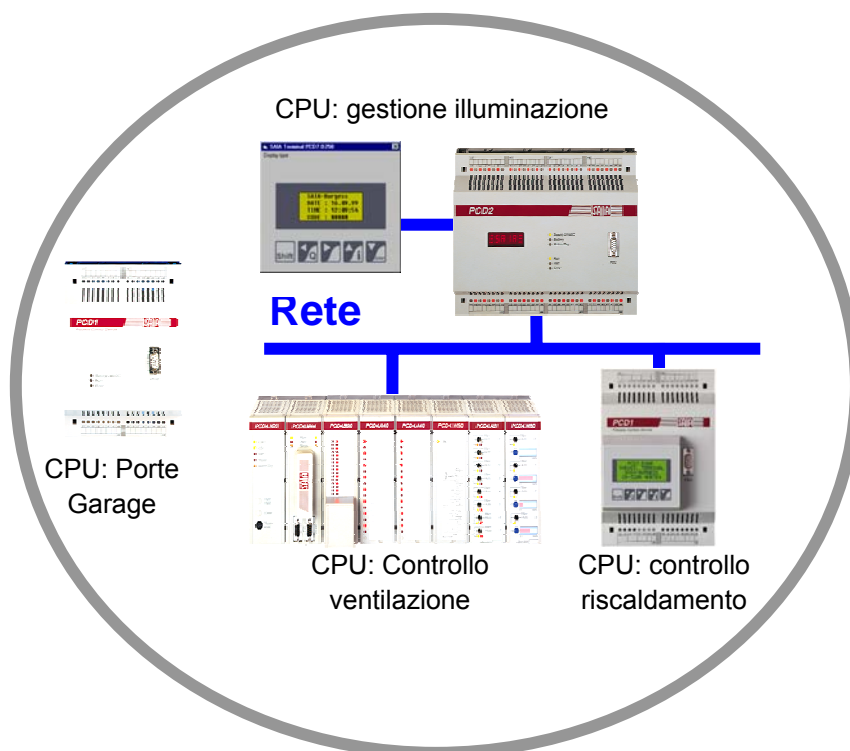
Le moderne applicazioni di automazione comprendono frequentemente un elevato numero di controllori collegati in rete. Per questo, il pacchetto PG5 riunisce in un singolo progetto i programmi e le configurazioni di tutti i controllori PCD previsti nell'ambito di una qualsiasi applicazione. Il Project Manager offre all'utente una visione globale di tutte le informazioni relative ad un progetto.

## 2.2 Organizzazione del progetto

### 2.2.1 Esempio di progetto

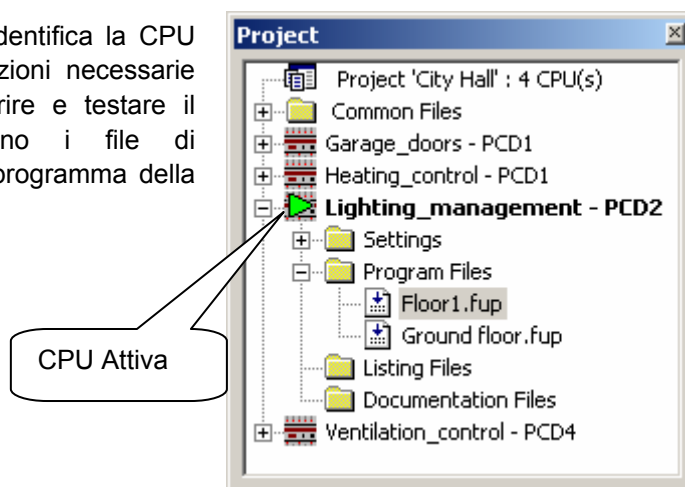
In pratica, gli impianti automatizzati comprendono quasi sempre un certo numero di controllori programmabili PCD locali collegati in una rete di comunicazione. Ciascun PCD supporta il controllo globale di una particolare funzione nell'ambito del progetto: ad esempio la gestione dell'illuminazione, il controllo del riscaldamento, il controllo della ventilazione, o le porte automatiche di un garage sotterraneo.

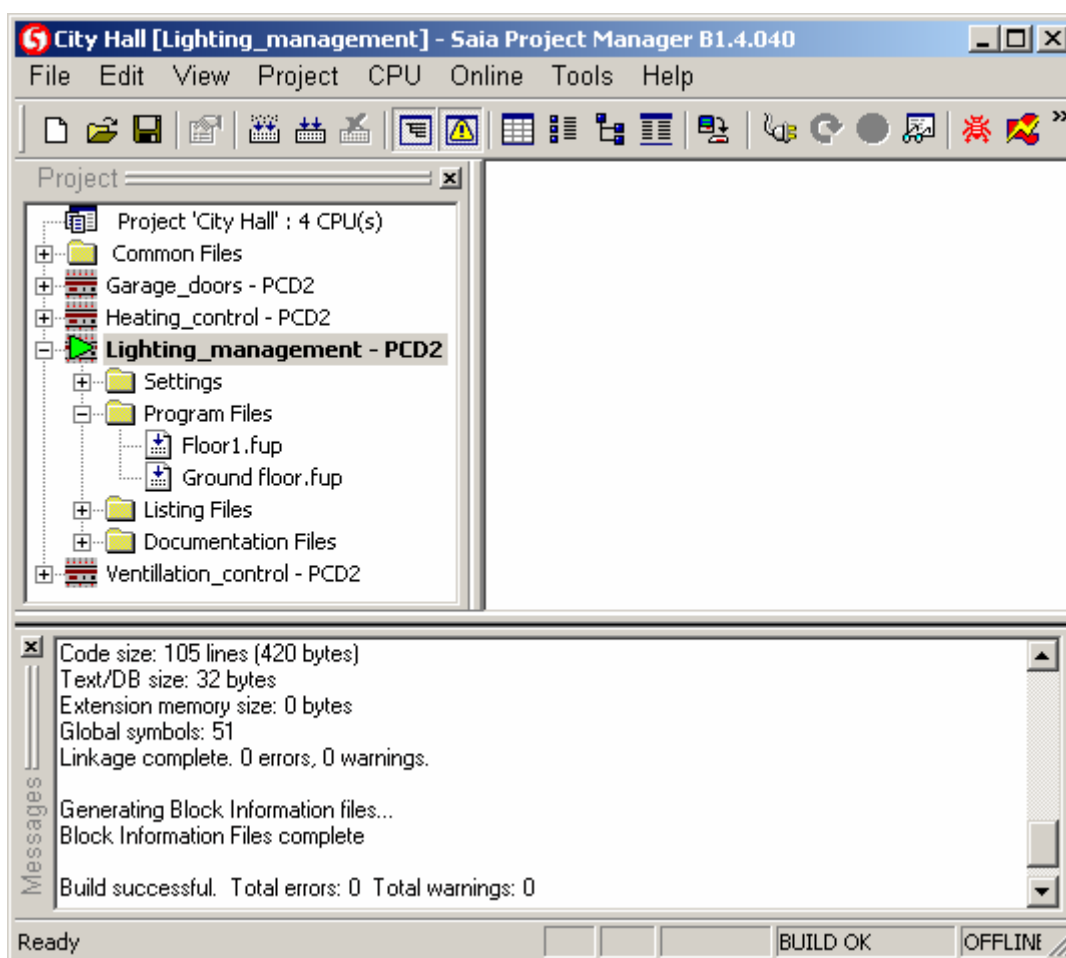
#### Progetto : City Hall (Municipio)



Il pacchetto software PG5 riunisce in un singolo progetto PG5 tutte le CPU dei PCD appartenenti ad una particolare applicazione.

Un triangolo verde identifica la CPU attiva. Tutte le istruzioni necessarie per costruire, trasferire e testare il programma utilizzano i file di configurazione e di programma della CPU attiva.





Struttura progetto

La finestra *Project (Progetto)* illustra la struttura del progetto con le CPU dei PCD che lo costituiscono. Per visualizzare questa finestra, selezionare il percorso di menu: *View, Project Tree (Visualizza, Struttura Progetto)*, oppure fare clic sul pulsante *Project Tree (Struttura Progetto)*.



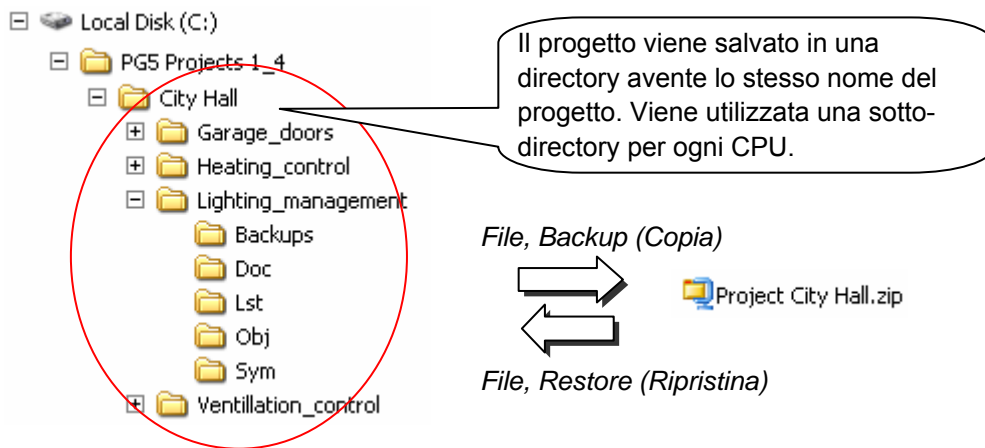
Finestra Messaggi

La finestra *Message (Messaggi)* visualizza i messaggi di allarme e di errore generati durante le fasi di costruzione del programma. Per visualizzare questa finestra, selezionare la voce di menu: *View, Message Window (Visualizza, Finestra Messaggi)*, oppure fare clic sul pulsante *Message Window (Finestra Messaggi)*.

La finestra *View (Vista)* visualizza l'elenco delle liste, l'elenco dei blocchi, la struttura dei blocchi o i file di testo. Inoltre permette di ottenere i riferimenti incrociati dei simboli.

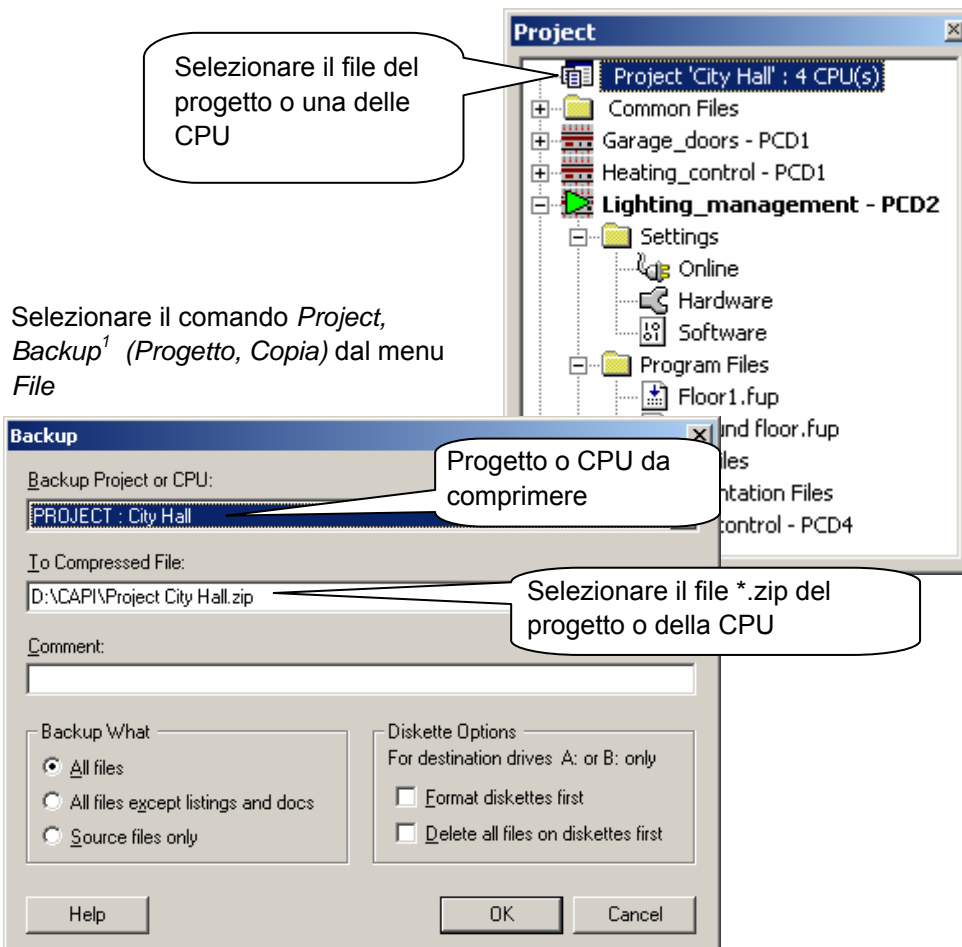
### 2.2.2 Salvataggio del progetto su PC

Per default, i progetti vengono salvati nella directory **C:\PG5 Projects 1\_4**



### 2.2.3 Compressione di un progetto o di una CPU

Quando un progetto viene salvato, deve essere preservata l'intera struttura delle directory e dei file in esse contenuti. Il metodo più semplice per fare ciò è quello di comprimere l'intera struttura del progetto in un file \*.ZIP utilizzando il comando Backup (Copia). L'operazione può essere effettuata nel modo seguente:

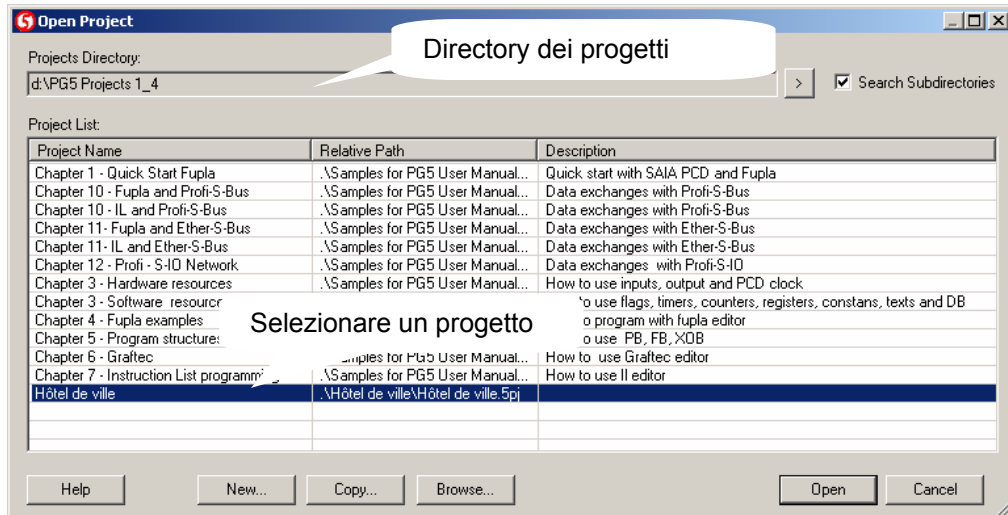


<sup>1</sup> Il comando *Project, Restore (Progetto, Ripristina)* del menu *File* permette di ripristinare un progetto o una CPU precedentemente salvati in un file \*.ZIP.

## 2.2.4 Apertura di un progetto

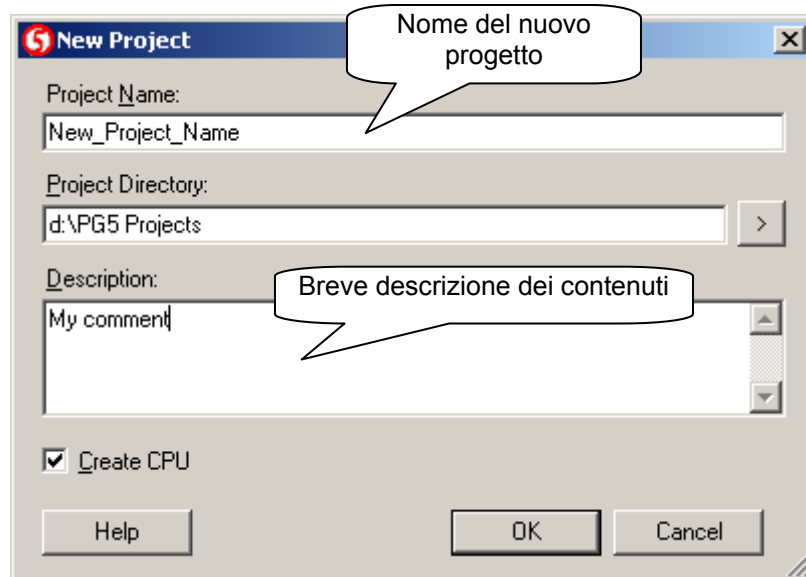
Il pacchetto PG5 viene rilasciato con tutti gli esempi riportati in questo manuale, inclusi. Il comando Project, Open... (Progetto, Apri) del menu File permette di aprire e verificare tali esempi.

Un progetto esistente può essere aperto con il comando Project, Open... (Progetto, Apri) del menu File. Il suddetto comando individua tutti i file di progetto (.5pj) contenuti nella directory dei progetti e li visualizza sotto forma di elenco. Fare doppio clic su uno dei progetti dell'elenco oppure selezionare un progetto e premere il pulsante Open (Apri). In alternativa, premere il pulsante Browse (Scorri) e scegliere direttamente il file di progetto (.5pj) o il file di CPU (.5pc).



## 2.2.5 Creazione di un nuovo progetto

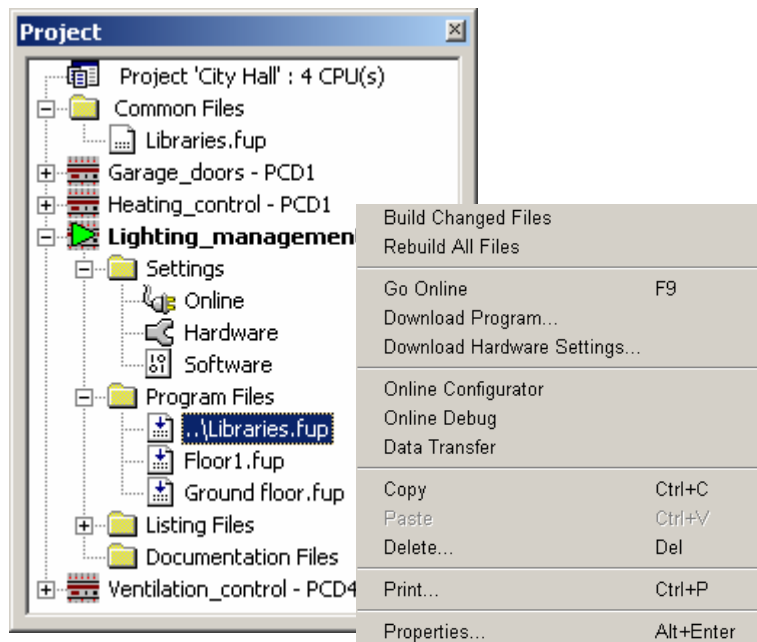
Per creare un nuovo progetto, utilizzare il comando *Project, New...* (Progetto, Nuovo) del menu *File*, definire il nome del nuovo progetto nel campo *Project Name* (Nome Progetto), selezionare l'opzione *Create CPU* (Crea CPU) e confermare con il pulsante *OK*.



## 2.3 Finestra *Project (Progetto)*



Albero dei  
Progetti



Le cartelle contenute nella finestra *Project (Progetto)* raggruppano le informazioni relative al progetto in base a determinati criteri organizzativi:

### 2.3.1 Cartella Progetto



La cartella principale mostra il progetto con il relativo nome e le CPU incluse nel progetto stesso. Per modificare le informazioni contenute in questa cartella, selezionarla con il mouse ed attivare il menu contestuale utilizzando il pulsante destro del mouse.

<i>New CPU... (Nuova CPU)</i>	Aggiunge una nuova CPU al progetto
<i>Import CPU... (Importa CPU)</i>	Importa CPU da un altro progetto PG5 o da un precedente progetto PG4
<i>Paste CPU (Incolla CPU)</i>	Incolla una CPU assieme a tutti i relativi contenuti
<i>Copy Project... (Copia Progetto)</i>	Copia un progetto e tutti i relativi contenuti
<i>Backup... (Comprimi)</i>	Esegue un backup del progetto all'interno di un file *.zip
<i>Restore... (Ripristina)</i>	Ripristina un progetto salvato sottoforma di file *.zip
<i>Rebuild All CPUs Online Commands (Costruisci Tutte le CPU Comandi In Linea)</i>	Comandi per effettuare la Costruzione (Build) o il trasferimento (Download) dell'intero progetto.
<i>Print...</i>	<i>(Stampa)</i>
<i>Find...</i>	<i>(Trova)</i>
<i>Properties (Proprietà)</i>	Modifica delle informazioni associate alla cartella del progetto: nome progetto, descrizione, ...

### 2.3.2 Cartella File Comuni

La cartella *Common Files (File Comuni)* contiene i moduli comuni a più CPU del progetto. Per aggiungere un file di programma, selezionare la cartella ed utilizzare l'opzione *New File (Nuovo File)* nel menu contestuale.


La voce *Add Files (Aggiungi File)* nel menu contestuale può essere utilizzata per importare qualsiasi tipo di file di programma PG5, ma può importare anche i documenti riguardanti la messa in servizio e la manutenzione dell'applicazione (in formato Word, Excel, ecc.). Questi file sono memorizzati nel progetto PG5 e possono essere aperti facendo doppio clic su di essi.

#### Nota

I file comuni utilizzano gli stessi *Simboli Locali* in ogni CPU che fa uso di tali file, ma vengono utilizzati anche i *Simboli Globali* propri della CPU, per cui i simboli globali utilizzati in un file comune possono essere diversi per ogni CPU.

### 2.3.3 Cartella CPU

Ogni cartella CPU contiene le configurazioni ed i programmi per un solo controllore utilizzato nel progetto. Per modificare le informazioni contenute in una cartella CPU, fare doppio clic su di essa per attivare il menu contestuale.

<i>Set Active (Rendi Attiva)</i> 	Questo comando permette di rendere attiva la CPU attualmente selezionata all'interno della Finestra <i>Project (Progetto)</i> . La CPU attiva viene identificata da un triangolino verde. Tutti i comandi impartiti via menu e pulsanti sono riferiti alla CPU attiva.
<i>Rebuild Changed files (Costruisci File Modificati)</i> <i>Rebuild All Files (Costruisci Tutti i File)</i>	Comandi per eseguire la "Costruzione" ( <i>Build</i> ) della CPU attualmente selezionata all'interno della Finestra <i>Project (Progetto)</i> .
<i>Go Online</i> <i>Download Program</i> <i>Download Hardware Settings</i>	( <i>Passa In Linea</i> ) ( <i>Trasferisci Programma</i> ) ( <i>Trasferisci Impostazioni Hardware</i> )
<i>Online Configurator</i> <i>Online Debug</i> <i>Data Transfer</i>	( <i>Configuratore In Linea</i> ) ( <i>Debug In Linea</i> ) ( <i>Trasferimento Dati</i> )
<i>Copy, Paste, Delete (Copia, Incolla, Cancella)</i>	Copia, incolla o cancella la CPU attualmente selezionata all'interno della Finestra <i>Project (Progetto)</i> .
<i>Print...</i>	( <i>Stampa</i> )
<i>Properties (Proprietà)</i>	Modifica delle informazioni associate alla cartella CPU: nome CPU, descrizione, ...

#### Nota:

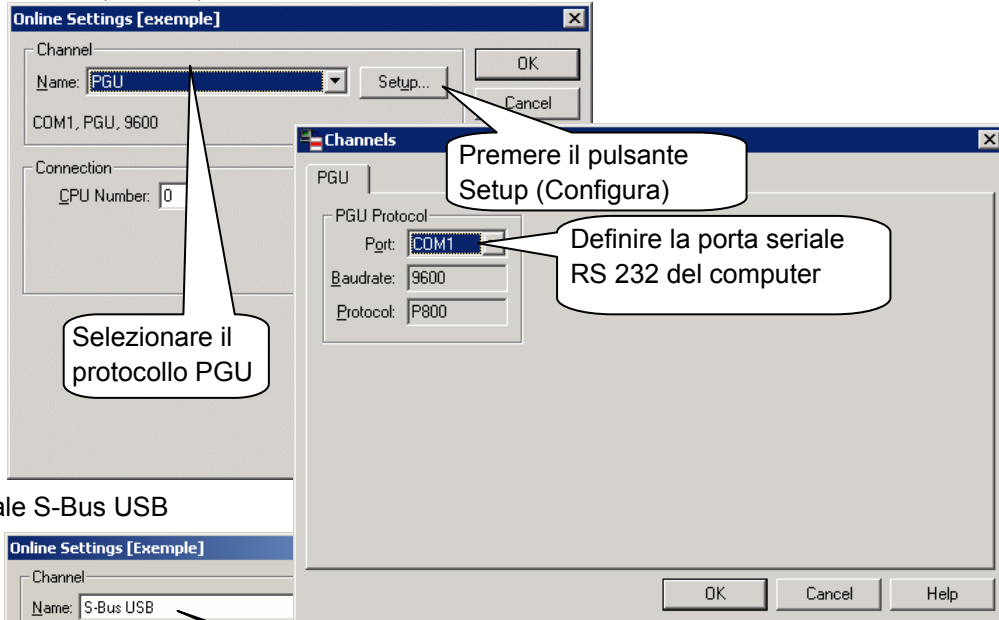
Se è attiva l'opzione *Activate CPU according to Project Tree location (Attiva CPU in base alla dislocazione nell'Albero del Progetto)* selezionabile mediante la voce di menu: *Tools, Options, General page (Strumenti, Opzioni, Generali)*, *Il Project Manager (Gestore Progetti) SAIA* attiverà automaticamente la CPU in base al contesto. In questo caso, non sarà possibile utilizzare il comando *Set Active (rendi Attiva)* e quest'ultimo non verrà pertanto visualizzato all'interno del menu contestuale.

### 2.3.4 Impostazioni In Linea (Online)

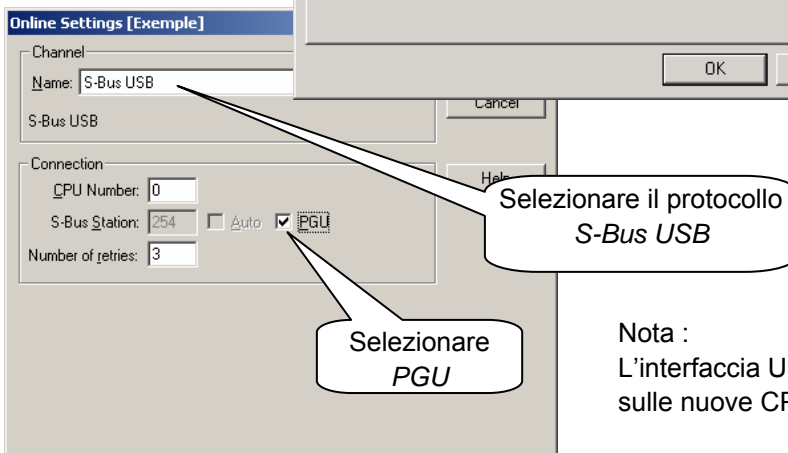


La cartella *Settings, Online (Impostazioni, In Linea)* permette di definire i parametri di comunicazione della CPU. Sono supportati vari protocolli di comunicazione: PGU, S-Bus, Ethernet, ecc. Tuttavia, solo i protocolli PGU ed S-Bus USB permettono di comunicare direttamente con il PCD senza richiedere la configurazione hardware.

#### Canale PGU (RS 232)



#### Canale S-Bus USB

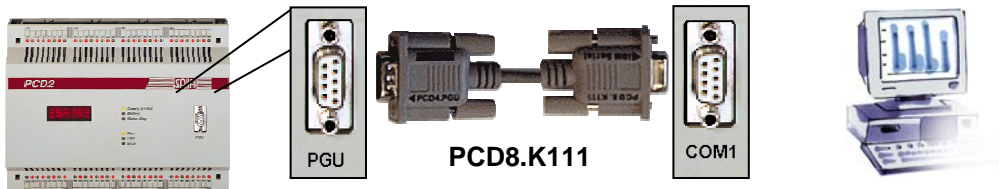


Nota :  
L'interfaccia USB è disponibile solo sulle nuove CPU PCD2.M480 e PCD3

### 2.3.5 Collegamento PC-PCD

#### Canale PGU (RS 232)

Il collegamento RS 232 tra PC e PCD è realizzato dal cavo PCD8.K111. Per ulteriori dettagli su questo cavo, fare riferimento al manuale hardware PCD.



#### Canale S-Bus USB

L'interfaccia USB è disponibile solo sulle nuove CPU PCD2.M480 e PCD3



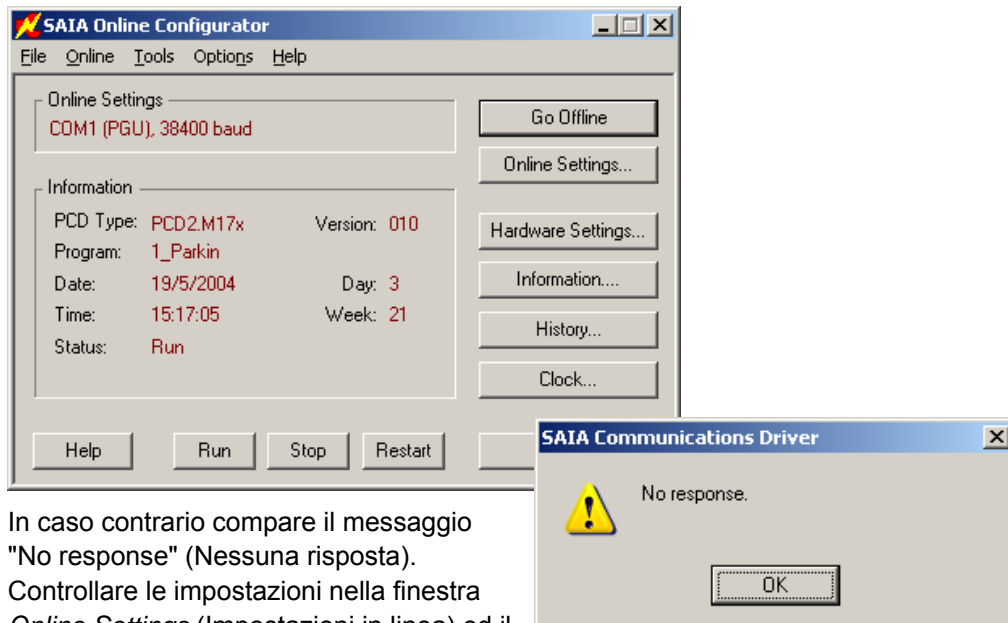




Configuratore  
In Linea

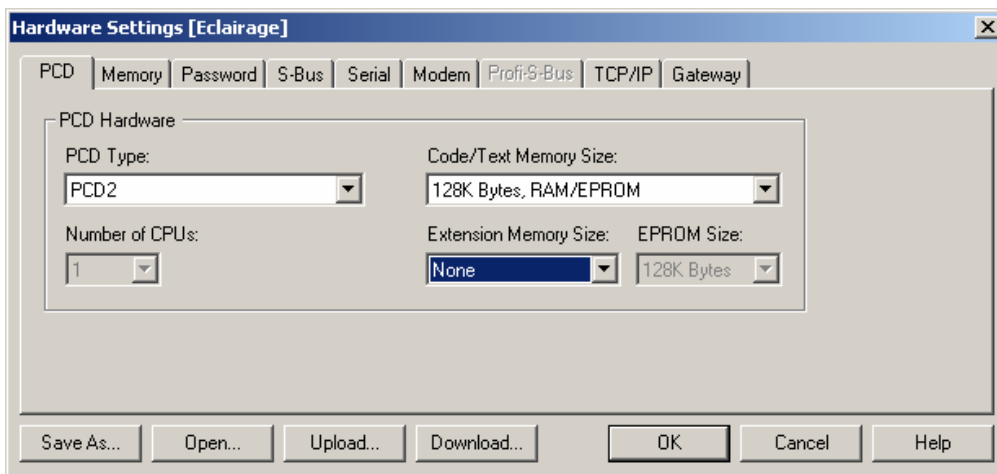
### Controllo della Connessione

Il pulsante *Online Configurator* (*Configuratore In Linea*), oppure il comando omonimo del menu *Tools* (*Strumenti*) permettono di visualizzare le impostazioni del PCD collegato. Se vengono visualizzate le informazioni indicate in rosso, significa che la comunicazione sta funzionando perfettamente.



In caso contrario compare il messaggio "No response" (Nessuna risposta). Controllare le impostazioni nella finestra *Online Settings* (Impostazioni in linea) ed il cavo di comunicazione.

### 2.3.6 Impostazioni Hardware




La cartella *Hardware Settings* (*Impostazioni Hardware*) permette di definire la memoria e i parametri di comunicazione del controllore PCD.

Quando un controllore viene utilizzato per la prima volta, oppure dopo l'aggiunta di nuova memoria al PCD, è necessario configurare la memoria. Esistono due modi per selezionare i parametri della finestra sopra illustrata:

- Il primo modo consiste nel selezionare il pulsante *Upload* (*Caricamento*) e leggere i parametri direttamente dal controllore.
- Il secondo modo consiste nel definire le informazioni della finestra con l'ausilio delle tabelle riportate nelle due pagine seguenti. L'esempio indicato corrisponde alle righe contrassegnate in grassetto nelle suddette tabelle.

La seguente tabella riporta le informazioni riguardanti i ponticelli di selezione memoria. Tali ponticelli devono essere configurati sulla scheda CPU del PCD. Per ulteriori informazioni, si prega di fare riferimento al manuale hardware.

Tipo di PCD	Memoria	Numero d'ordine	Dimensione memoria Istr./Testi	Estensione di Memoria (RAM)	Posizione ponticello su PCD
PCS1.C8	Flash 2 MBit		240 kByte	128 kByte	Nessuna
	Flash 4 MBit		<sup>3)</sup> 1008 kByte	896 kByte	Nessuna
PCD1. M110 /120/150	Spazio vuoto		17 kByte	No	R
	1 RAM 256 kBit	4 502 5414 0	32 kByte	13 kByte	R
	1 RAM 1MBit	4 502 7013 0	128 kByte	13 kByte	R
	1 Flash 1MBit	4 502 7141 0	112 kByte	13 kByte	E
	1 EPROM 512kBit	4 502 3958 0	64 kByte	13 kByte	E
	1 EPROM 1Mbit	4 502 7126 0	128 kByte	13 kByte	E
PCD2.M110 /120/150 	Spazio vuoto		32/128 kByte <sup>1)</sup>	Nessuna	R , <=1Mbit
	1 RAM 256 kBit	4 502 5414 0	32 kByte	24/128 kByte <sup>1)</sup>	R , <=1Mbit
	<b>1 RAM 1 Mbit</b>	4 502 7013 0	<b>128 kByte</b>	<b>24/128 kByte <sup>1)</sup></b>	<b>R , &lt;=1Mbit</b>
	1 RAM 4 Mbits	4 502 7175 0	512 kByte	24/128 kByte <sup>1)</sup>	R , >1Mbit
	1 Flash 1 MBit	4 502 7141 0	112 kByte	24/128 kByte <sup>1)</sup>	F , <=1Mbit
	1 Flash 4 MBit	4 502 7224 0	448 kByte	24/128 kByte <sup>1)</sup>	F , >1Mbit
	1 EPROM 512 kBit	4 502 3958 0	64 kByte	24/128 kByte <sup>1)</sup>	E , <=1Mbit
	1 EPROM 1 MBit	4 502 7126 0	128 kByte	24/128 kByte <sup>1)</sup>	E , <=1Mbit
	1 EPROM 4 MBit	4 502 7223 0	512 kByte	24/128 kByte <sup>1)</sup>	E , >1Mbit
PCD2.M170	RAM 1 MBit		1024 kByte		Nessuna
PCD2.M480	RAM 1 MBit		1024 kByte		Nessuna
PCD3.M3020 PCD3.M3230	RAM + Flash di backup a bordo 256 KB		<sup>2)</sup> 256 kByte	No	Nessuna
PCD3.M5440 PCD3.M5540	RAM 512 KB + Flash di backup a bordo 256 KB		<sup>2)</sup> 256 kByte	No	Nessuna
	RAM 512 KB + Flash di backup 512 KB	PCD7.R500	<sup>2)</sup> 512 kByte	No	Nessuna
PCD4.M	2 RAM 62256	4 502 5414 0	64 kByte	172 kByte, per memoria PCD7.R310	RAM
	2 RAM 1 Mbit	4 502 7013 0	256 kByte		RAM
	2 EPROM 256 kBit	4 502 5327 0	64 kByte		E256
	2 EPROM 512 kBit	4 502 3958 0	128 kByte		E512
	2 EPROM 1 MBit	4 502 7126 0	256 kByte		E1M
PCD4.M170	RAM 1 MBit	PCD4.M170	1024 kByte		Nessuna

<sup>1)</sup> 128 kByte per tutti i PCD2.M110/120 con versione hardware J o superiore, 128 kByte per tutti i PCD2.M150

<sup>2)</sup> Dimensione memoria flash per backup

<sup>3)</sup> 1008 kByte per tutti i PCS1 versione hardware E o superiore

Ponticelli::R=RAM,E=EPROM,F=Flash

Se la dimensione della memoria non è nota, rilevare il codice di riferimento stampato sul componente di memoria ed utilizzare la tabella seguente per stabilire il numero d'ordine e la relativa dimensione di memoria:

Capacità del Componente	Numero di Ordine	Codice di Riferimento
RAM 256 kBit	4 502 5414 0	SRM 2B256SLCX70 HY62256ALP-70 GM76C256CLL-70 M5M5256DP-70LL TC55257DPL-70L
RAM 1 MBit	4 502 7013 0	BS62LV1025 PC-70 LP621024D-70LL SRM20100LLC70 HY628100ALP-70 GM76C8128CLL-70 M5M51008BP-70L TC551001BLP-70L
RAM 4 MBit	4 502 7175 0	BS62LV4006PC P55 BS62LV4007PC P55 HM628512LP-5 KM684000ALP-5L KM684000BLP-5L
Flash 1 MBit	4 502 7141 0	AM29F010-70PC
Flash 4 MBit	4 502 7224 0	AM29F040 (Socket)
EPROM 256 kBit	4 502 5327 0	UPD27C256AD-10 M27C256B-10F1 TMS27C256-10JL
EPROM 512 kBit	4 502 3958 0	AM27C512-15XF1 AMC27C512-15XF1 AM27C512-90DC UPD27C512D-10 M27512-10XF1 M27512-10F1
EPROM 1 MBit	4 502 7126 0	AM27C010-90DC NM27C010Q-90 M27C1001-10F1
EPROM 4 MBit	4 502 7223 0	AM27C040-100DC M27C4001-10F1

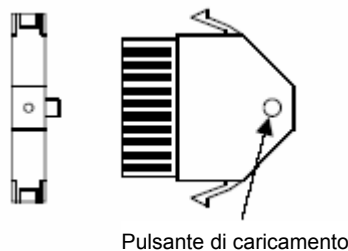
Tipo di PCD	Estensione di Memoria Istruzioni/Testi (RAM)	Memoria di Backup Interna (Flash )	Memoria di Backup Esterna (Flash )
PCD4.M170 PCD2.M170 PCD2.M480	1024 kByte	Nessuna	PCD7.R400 - 1024 kByte
PCD3.M3020P CD3.M3120	128 kByte	128 kByte	PCD3.R500 - 128 kByte
PCD3.M3230 PCD3.M3330	256 kByte	256 kByte	PCD3.R500 - 256 kByte
PCD3.M5440 PCD3.M5540 PCD3.M6340 PCD3.M6540	512 kByte	256 kByte	PCD7.R500 - 512 kByte

I nuovi sistemi PCD supportano la memoria di backup interna o esterna PCD7.R400/R500 (opzionale).

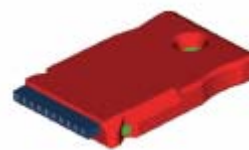
La memoria di backup permette di salvare una copia del programma applicativo (istruzioni/testi/estensioni) attualmente memorizzato sulla RAM all'interno di una memoria flash, il cui contenuto non verrà perso in caso di caduta di tensione o malfunzionamento/scarica della batteria.

Si raccomanda pertanto di utilizzare la suddetta memoria di backup sui sistemi PCD al fine di assicurarsi una protezione contro qualsiasi perdita di dati indesiderata.

Se il programma applicativo in RAM (istruzioni/testi/estensioni) è corrotto, eseguendo una ripartenza a freddo del PCD si otterrà automaticamente il ripristino del programma dalla memoria di backup.



PCD7.R400

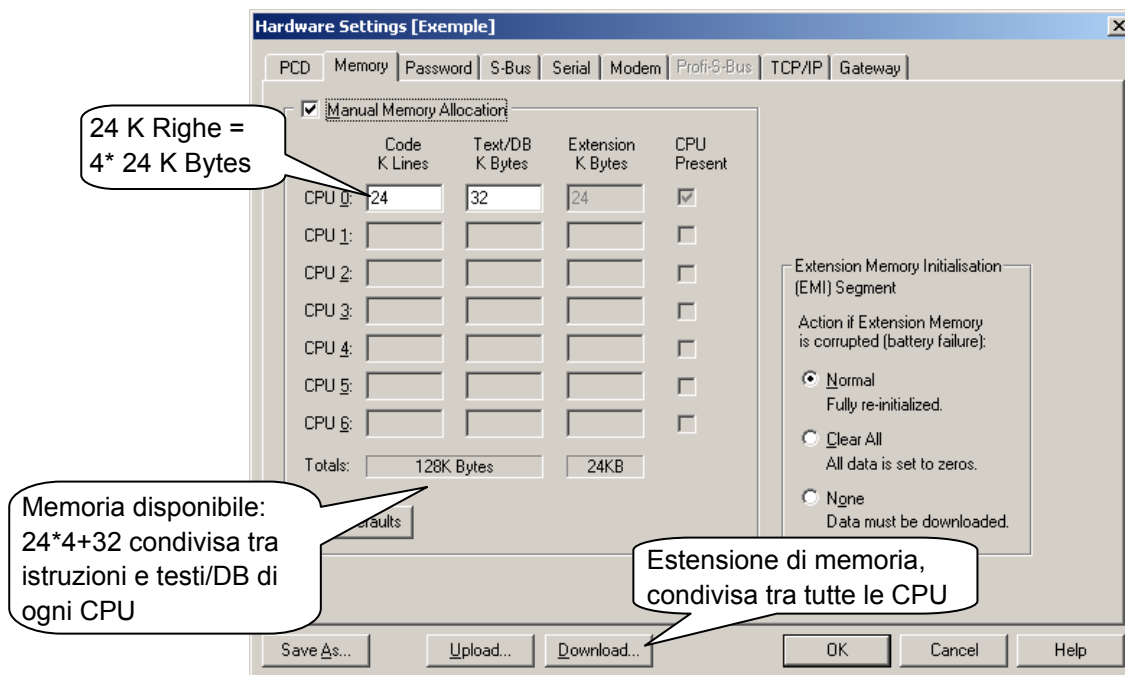


PCD7.R500

La memoria di backup esterna permette inoltre di trasferire applicazioni da un PLC all'altro e di creare una copia dei testi e dei DB in RAM, nella memoria di estensione con PLC in funzionamento (indirizzo  $\geq 4000$ ).

Nota:

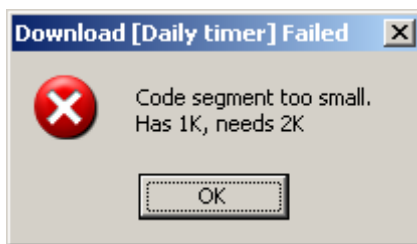
Indipendentemente dalla memorizzazione sulla memoria di backup, è comunque necessario eseguire sempre un backup dei file sorgenti del progetto. Questi ultimi non sono infatti memorizzati nella memoria del PCD.



La memoria disponibile definita nella pagina PCD è condivisa tra istruzioni di programma e testi in ogni CPU. Alcuni PCD hanno più di una CPU: PCD4.M44x e PCD6.Mxxx.

In caso di CPU singola questo viene definito automaticamente in base al programma utente, quindi la casella *Manual Memory Allocation (Allocazione Memoria Manuale)* può rimanere non selezionata.

I parametri di default sono adeguati per la maggior parte delle applicazioni. Nelle applicazioni in cui tali parametri di default non sono adeguati, compare un messaggio di errore simile a quello qui di seguito raffigurato quando si tenta di trasferire il programma sul PCD:



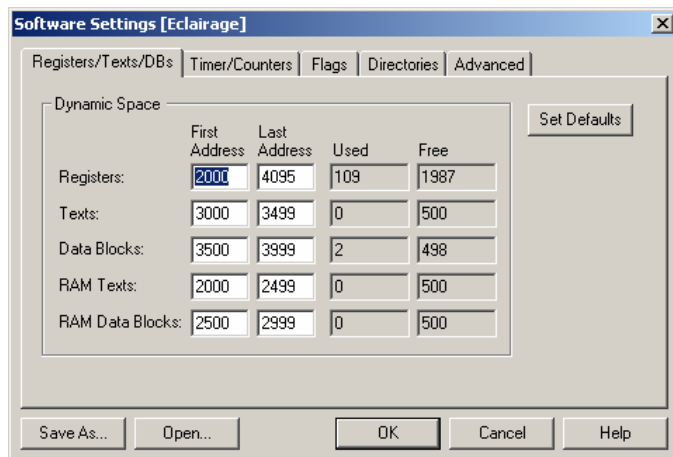
Questo tipo di errore può essere risolto in vari modi:

- Non selezionare l'opzione *Manual Memory Allocation (Allocazione Memoria Manuale)* lasciando al software PG5 il compito della ripartizione istruzioni/testo, purché la quantità di memoria sia sufficiente.
- Selezionare l'opzione *Manual Memory Allocation (Allocazione Memoria Manuale)* e configurare l'allocazione di memoria in base alle indicazioni del messaggio di errore.
- Incrementare la capacità di memoria del PCD.



Dopo aver definito le *Impostazioni Hardware (Hardware Settings)*, ricordarsi sempre di trasferirle nel PCD premendo il pulsante *Download (Trasferisci)* oppure utilizzando il comando *Download* nel menu *Online, Hardware Settings (In Linea, Impostazioni Hardware)*.

## 2.3.7 Impostazioni Software



Questa finestra permette all'utente di riservare dei campi di indirizzi per registri, contatori, temporizzatori e flag dinamici. Durante la costruzione del programma, questi indirizzi vengono automaticamente assegnati a simboli dinamici dal programma utente e dagli FBox Fupla.

Un simbolo dinamico è un simbolo per il quale non è stato definito alcun indirizzo hardware:

Simbolo Dinamico				
HMS	R			PCD Clock with current time
DailyTimer	Output	32		Daily Timer

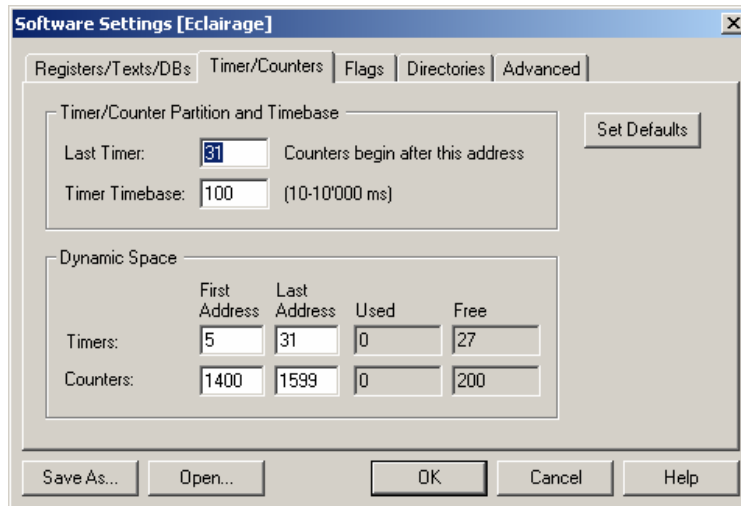
Non è sempre necessario modificare gli indirizzi dinamici. Le impostazioni di default sono normalmente adeguate per la maggior parte delle applicazioni.

Tuttavia, nel caso in cui durante la costruzione di un programma di grandi dimensioni compaia un messaggio del tipo:

**Fatal Error 368: Auto-allocation/dynamic space overflow for type: R (Errore Fatale 368: Superamento dello spazio ad allocazione automatica/dinamica per il tipo : R)**

sarà necessario estendere il campo di indirizzi per il tipo di elemento indicato nel messaggio di errore.

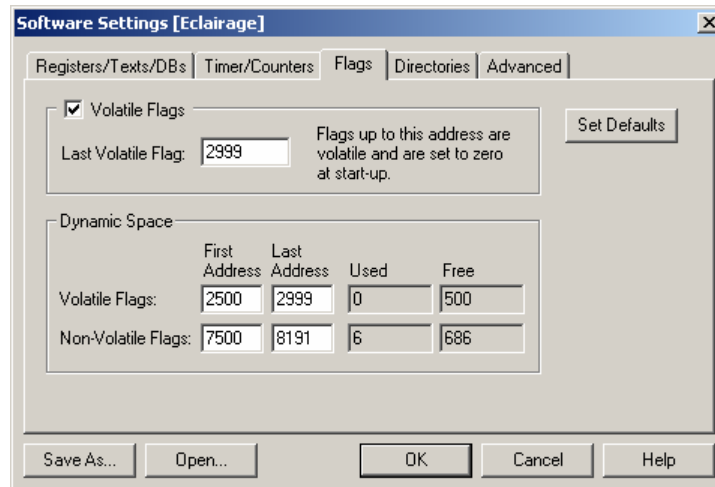
Se il controllore è equipaggiato con memoria di tipo EPROM o Flash, dovranno essere configurati anche i campi dinamici *RAM Texts (Testi RAM)* e *RAM Data Block (Data Block RAM)* a partire dall'indirizzo 4000, in modo che tali testi e DB possano essere scritti in memoria RAM.



I PCD sono configurati con 31 temporizzatori, alcuni dei quali hanno gli indirizzi assegnati dinamicamente. Con determinati programmi può essere necessario aumentare il numero di temporizzatori.

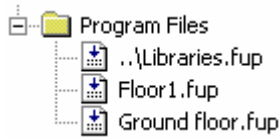
La base tempi utilizzata per decrementare i contatori è di 0,1 secondi (100ms). Se necessario, può comunque essere impostato un altro valore. Occorre in ogni caso notare che la base tempi non ha alcuna influenza sui programmi Fupla. Solo i programmi IL sono influenzati da questo parametro.

È consigliabile non definire un numero eccessivamente elevato di temporizzatori, e neppure una base tempi troppo piccola, se non strettamente necessario. Questo contribuirà a velocizzare i tempi di ciclo del programma.



Per default, tutti i flag sono non volatili. Se necessario, si può tuttavia utilizzare il parametro *Last Volatile Flag (Ultimo Flag Volatile)* per definire un intervallo di flag volatili. (L'esempio illustrato in questa videata definisce come volatili i flag da F 0 a F 2999.) I flag volatili vengono sempre impostati a 0 all'avviamento, mentre i flag non volatili, mantengono i valori precedenti.

### 2.3.8 Cartella File di Programma (Program Files)



Questa cartella contiene i file che costituiscono il programma della CPU. Per modificare i file in una cartella di programmi, fare clic con il pulsante destro del mouse sul nome della cartella o del file, per attivare il menu contestuale.

Aggiungere un nuovo file di programma  
 Importa un file da un altro progetto o CPU  
 Copia/Incolla file di programma tra le varie CPU del progetto  
 Permette di visualizzare o modificare il nome, la descrizione e l'opzione di collegamento del file  
 Collegato o non collegato

Open	Enter
New...	Ctrl+N
Add Files...	
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Delete...	Del
✓ Linked	Ctrl+L
Properties...	Alt+Enter

Per aggiungere un nuovo file alla cartella programmi, occorre definire il nome e il tipo di file.

Nome file  
 Definire il tipo di editore in base all'elenco  
 Breve descrizione del modulo

Linked/Built     Open file now

Help    OK    Cancel



### 2.3.9 Tipi di file

Una CPU può avere vari file di programma di tipi diversi. Ogni tipo di file ha un corrispondente editore, specifico per determinati campi di applicazione.

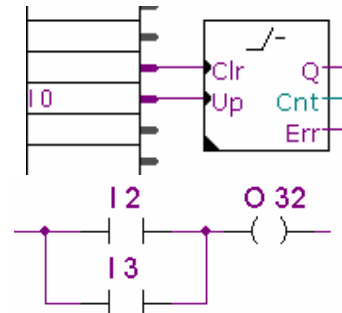
#### Editore IL (Lista Istruzioni) (\*.src)

Permette di scrivere i programmi sotto forma di testi, utilizzando un repertorio di 127 istruzioni. È adatto per tutte le applicazioni, ma richiede una certa esperienza di programmazione.

```
COB 0
      0
STH  I 0
DYN  F 9
INC  C 53
ECOB
```

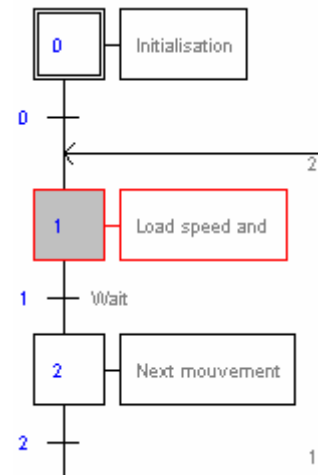
#### Editore Fupla (\*.fup)

Permette di disegnare i programmi sotto forma di schemi funzione e diagrammi di contatti. Non richiede esperienze di programmazione specifiche. Sono disponibili svariate librerie per la rapida implementazione di applicazioni HEAVAC e reti di comunicazione (modem, Lon, Belimo, EIB, ecc.).



#### Editore Graftec (\*.sfc)

È uno strumento che permette di strutturare programmi scritti in IL (Lista Istruzioni) e Fupla. È particolarmente adatto per realizzare applicazioni sequenziali basate su attese di eventi interni ed esterni.



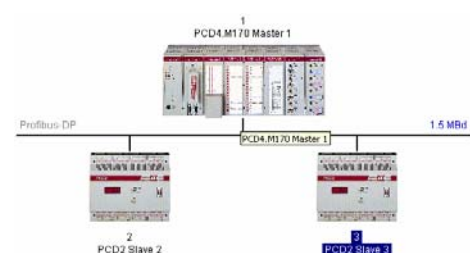
#### Editore HMI

Permette di configurare il dialogo con i terminali PCD7.D1xx e PCD7.D2xx (viene installato in aggiunta al pacchetto PG5)

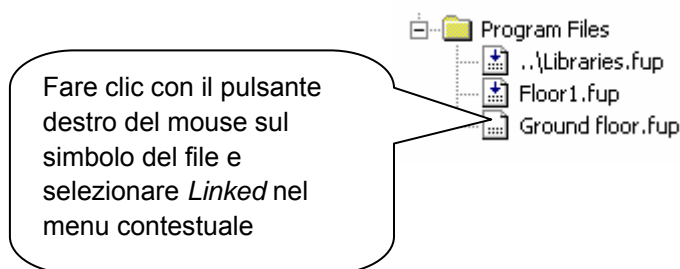


#### Editore S-Net (\*.dp, \*.lon, \*.rio)

Supporta la configurazione delle reti di comunicazione: Profibus DP, LON e SRIO.



### 2.3.10 File Collegati (*Linkati*)

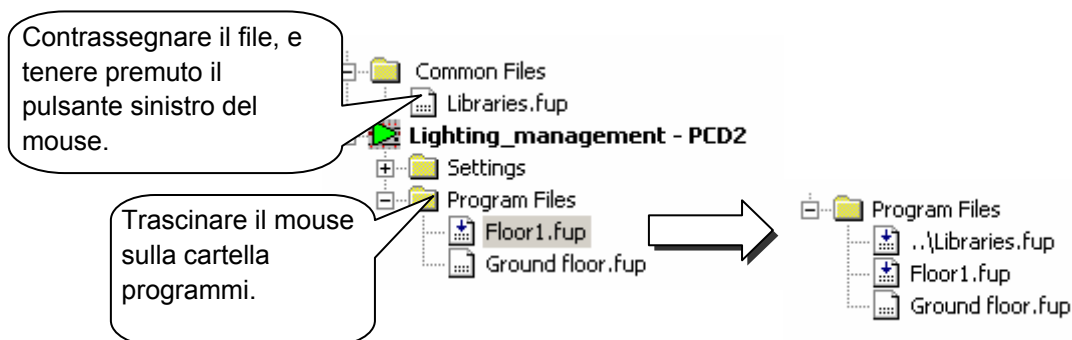


I file rappresentati da questa icona con il simbolo della freccia sono collegati (linkati) insieme per formare il programma e vengono trasferiti nella memoria del PCD.



I file rappresentati da questa icona senza il simbolo della freccia non fanno parte del programma. Questi file sono ignorati e non vengono trasferiti nella memoria del PCD. Questo può essere utile per i moduli che vengono collegati (linkati) solo per le attività di test durante la messa in servizio, ma che non devono essere inclusi nel programma finale.

### 2.3.11 File Comuni (*Common files*)



I file inclusi nella cartella *Common Files (File Comuni)* possono essere copiati, incollati o semplicemente trascinati nella cartella programmi della CPU che li utilizza. Notare i due punti all'inizio dei nomi file copiati o trascinati. Questi indicano che il file si trova in una cartella di livello superiore.

Il file può essere editato dalla cartella dei file comuni, oppure dalla cartella dei file di programma della CPU. In entrambi i casi, l'utente modifica lo stesso file e le correzioni verranno applicate a tutte le CPU collegate a tale file.

## 2.4 Costruzione del programma

Il PCD non può elaborare i programmi immediatamente dopo la fase di editazione in Fupla, IL, Graftec, S-Net o HMI. I file devono prima essere preparati seguendo le varie fasi illustrate in questo schema:

### File sorgenti:

- Programmi grafici



1. *Compilazione*

- Programmi in Lista Istruzioni



2. *Assemblaggio*



*Ricostruisci tutto o Costruisci*

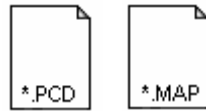
### File oggetto e listato



Stampante

3. *Link*

### File PCD

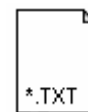


4. *Comando CPU, Crea Documentazione*



*Trasferimento programma*

### Controllore PCD



Stampante

1. La fase di compilazione converte i file grafici in file IL (\*.fdb, \*.src, \*.hsr)
2. La fase di assemblaggio produce i file in formato oggetto (\*.obj), ed un listato di assemblaggio (\*.lst) che può essere stampato o utilizzato per risolvere determinati errori dell'assembler.
3. La fase di Link combina i file oggetto (\*.obj) per formare un unico file eseguibile (\*.pcd) trasferibile nel controllore.
4. La documentazione può essere generata con il comando *Create Documentation (Crea documentazione)* nel menu *CPU* del Project Manager. Il risultato di questa operazione sarà disponibile nella cartella *Documentation Files (File di Documentazione)*.

### 2.4.1 Ricostruisci Tutto o Costruisci (Rebuild All e Build)



Ricostruisci tutto

La voce di menu *CPU, Rebuild all Files (CPU, Ricostruisci Tutti i File)* e/o il pulsante dedicato avviano la fase di compilazione, assemblaggio e link di tutti i file relativi alla CPU attiva.

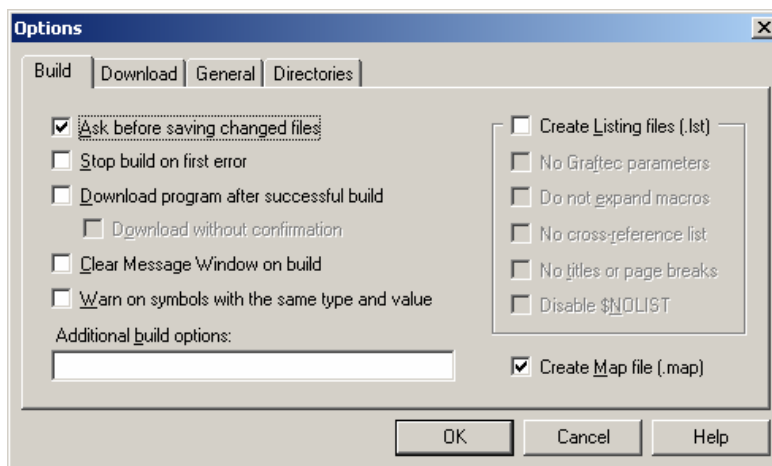


Costruisci File Modificati

La voce di menu *CPU, Build Changed Files (CPU, Costruisci File Modificati)* e/o il pulsante dedicato eseguono la stessa operazione, ma solo per i file che sono stati modificati dall'ultimo comando *Build Changed Files (Costruisci File Modificati)* o *Rebuild All Files (Ricostruisci Tutti i File)*. Questo permette di risparmiare tempo nella costruzione dei programmi di grandi dimensioni.

### 2.4.2 Opzioni della funzione di "Costruzione" (Build)

Con il comando *Options (Opzioni)* del menu *Tools (Strumenti)* si possono impostare varie opzioni per personalizzare il processo di costruzione programmi:



**Ask before saving changed files before a build** (*Richiesta prima di salvare i file modificati*)

Se si seleziona questa opzione, il pacchetto PG5 richiede l'autorizzazione a salvare i file sorgenti modificati, ma non ancora salvati prima della costruzione del programma. In caso contrario, i file verranno salvati automaticamente.

**Stop build on first error** (*Arresto del processo di costruzione al primo errore*)

Se si seleziona questa opzione, verrà arrestato il processo di costruzione alla comparsa del primo errore nella finestra *Messaggi*.

**Download program after successful build** (*Trasferimento programma dopo una costruzione terminata con successo*)

Se si seleziona questa opzione, il programma verrà automaticamente trasferito nel PCD, ma solo se il processo di costruzione è terminato con successo.

**Download without confirmation** (*Trasferimento senza conferma*)

Normalmente, il processo di trasferimento del programma nella memoria del PCD inizia con una finestra di dialogo che comunica all'utente l'inizio della fase di trasferimento e che deve essere confermata con il pulsante OK. Se invece si seleziona questa opzione, verrà direttamente trasferito il programma, senza visualizzare la finestra di dialogo.

**Clear Message Window on build** (*Cancellazione della finestra messaggi durante la costruzione*)

La finestra *Messaggi* verrà cancellata all'inizio di ogni processo di costruzione.

**Create listing file** (*Creazione del file di listato*).

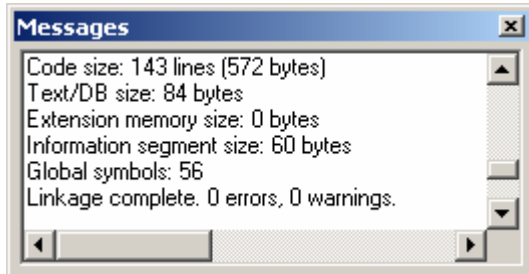
Crea un file contenente il listato del programma assemblato (\*.lst)

**Create map file** (*Creazione del file "mappa"*)

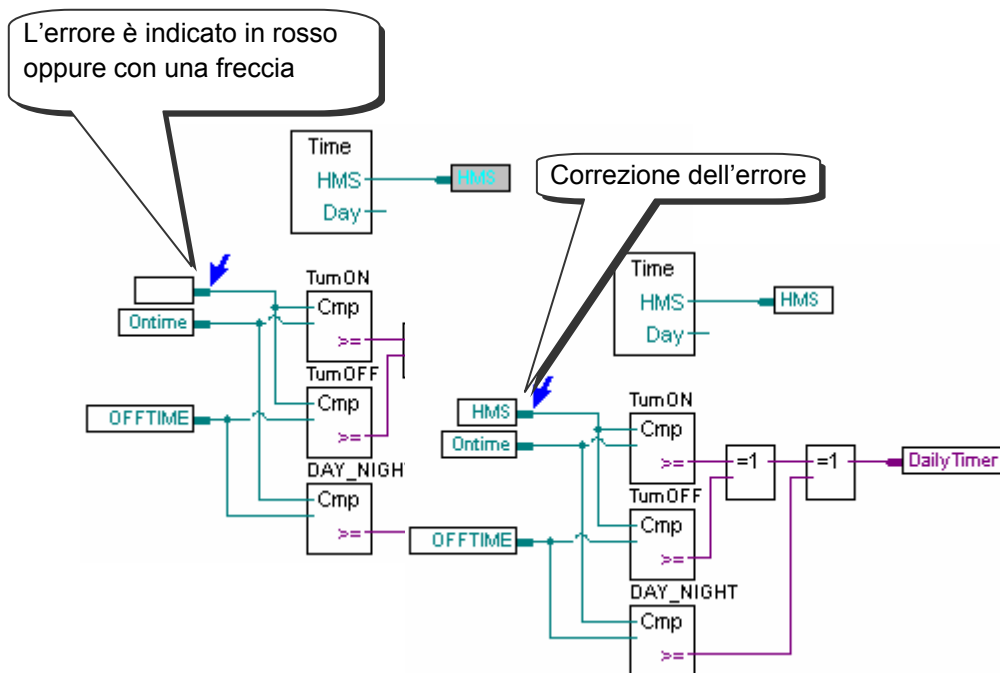
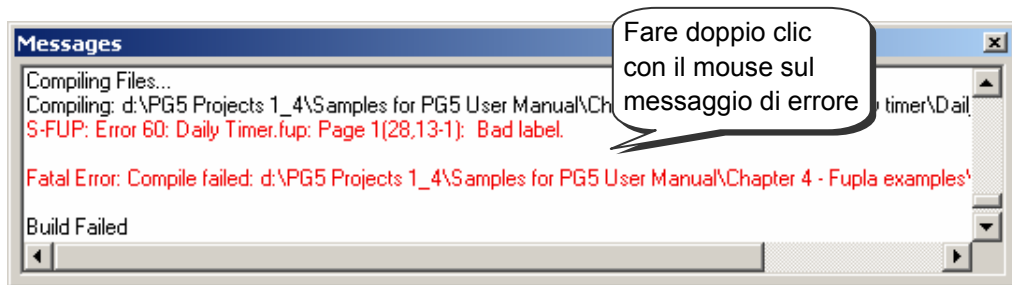
Crea un file contenente l'indicazione dello spazio di memoria occupato dall'applicazione e l'elenco dei simboli globali.

## 2.5 Finestra Messaggi (Messages)

La finestra *Messages (Messaggi)* fornisce le informazioni sullo stato di avanzamento del processo di costruzione di un programma. In particolare, evidenzia le varie fasi del processo: compilazione, assemblaggio e link. Se il programma è stato editato correttamente, il processo di costruzione termina con il messaggio: *Build successful. Total errors 0 Total warnings: 0 (Costruzione Terminata con successo, Totale Errori: 0, Totale Avvertimenti: 0)*



Gli eventuali errori verranno indicati con un messaggio scritto in rosso. Facendo doppio clic con il mouse su tale messaggio si abilita generalmente la localizzazione dell'errore nel contesto del programma applicativo.

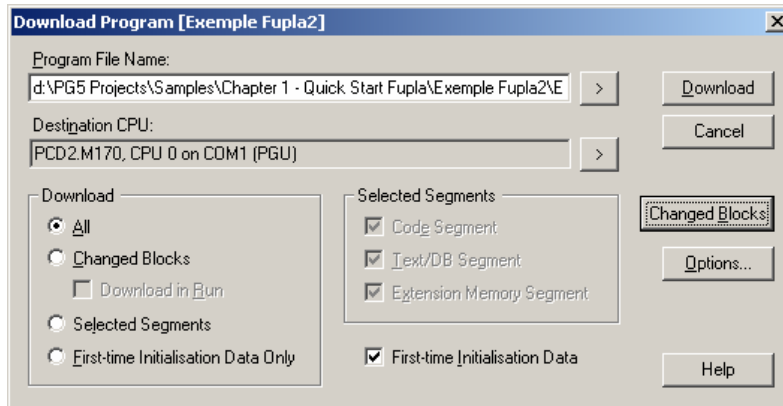


## 2.6 Trasferimento del Programma nel PCD (*Download Program*)



### Trasferimento Programma

Se il processo di costruzione si conclude senza messaggi di errore, si può utilizzare il pulsante, *Download Program (Trasferimento Programma)* o il comando omonimo nel menu *Online (In Linea)* per caricare il programma nella memoria del PCD.



#### **Program File Name** (*Nome del File di Programma*)

Per default è il nome del programma per la CPU attiva.

#### **All** (*Tutto*)

Trasferisce l'intero programma (*Code Segment (Segmento Istruzioni)*, *Text/DB Segment (Segmento Testi/DB)*, *Extension Memory Segment (Segmento Memoria di Espansione)*)

#### **Changed Blocks** (*Blocchi Modificati*)

Trasferisce solo i blocchi (COB,PB,FB,SB,ST,TR,XOB) modificati dopo l'ultima operazione di *Download*. Questa opzione viene normalmente utilizzata solo per risparmiare tempo in caso di piccole correzioni apportate al programma. Per visualizzare l'elenco dei blocchi modificati si può utilizzare il pulsante *Changed Blocks (Blocchi Modificati)*.

#### **Download in Run** (*Trasferimento durante l'esecuzione*)

Permette di trasferire i blocchi di programma modificati senza arrestare l'esecuzione del programma. Il corretto funzionamento di questa opzione può dipendere dalle correzioni apportate al programma.

#### **Selected Segments** (*Segmenti Selezionati*)

Trasferisce solo i segmenti definiti nella sezione *Selected Segments (Segmenti Selezionati)*:

*Code Segment* = Programmi, *Text/DB Segment* = Testi e DB da 0 a 3999,

*Extension Memory Segment* = Testi e DB da 4000 a 7999.

#### **First-time Initialisation Data Only** (*Solo i dati della prima inizializzazione*)

Trasferisce solo i dati descritti nel punto seguente.

#### **First-time Initialisation Data** (*Dati della prima inizializzazione*)

Questa opzione autorizza l'inizializzazione di certi Dati durante la costruzione di un programma. I dati inizializzati in fase di trasferimento programma sono definiti nel modo seguente:

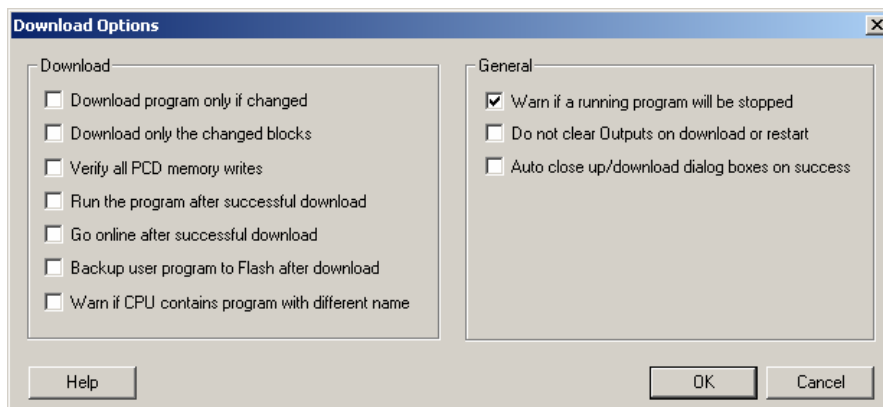
*indirizzo del tipo di simboli := valore\_inizializzazione*

Group/Symbol	Type	Address/Value	Comment
Symbol0	R	10:= 314	First time initialisation value = 314
Symbol1	R	11	

I dati non inizializzati durante il trasferimento del programma possono essere inizializzati ad ogni partenza a freddo dalle istruzioni contenute in XOB16.

### 2.6.1 Opzioni di Trasferimento (*Download options*)

Le opzioni di trasferimento possono essere definite con il comando *Options (Opzioni)* del menu *Tools (Strumenti)*, oppure con il pulsante omonimo nella finestra di dialogo *Download Program (Trasferimento Programma)*. Tali opzioni permettono di personalizzare la procedura di scaricamento programma.



***Download program only if changed*** (*Trasferimento dei soli programmi modificati*)

Trasferisce i programmi solo se sono stati modificati dopo l'ultimo download (*Trasferimento Programma*).

***Changed Blocks and Download in Run*** (*Solo blocchi modificati e trasferimento durante l'esecuzione*)

Trasferisce i programmi solo se sono stati modificati dopo l'ultimo download (*Trasferimento Programma*) durante l'esecuzione del programma.

***Download only the changed Blocks*** (*Trasferimento dei soli blocchi modificati*)

Vedere la pagina precedente.

***Verify all PCD memory writes*** (*Verifica di tutti i dati scritti nella memoria del PCD*)

Tutti i dati scritti nel PCD verranno riletti e confrontati a scopo di verifica. Questa opzione non dovrebbe normalmente essere selezionata, in quanto raddoppia il tempo richiesto per il trasferimento del programma.

***Run the program after successful download*** (*Esecuzione del programma dopo la corretta conclusione del trasferimento*)

Forza automaticamente la CPU in stato Run dopo il trasferimento di un programma. Attenzione: questa opzione deve essere selezionata solo quando esiste la certezza che il programma funziona correttamente e non sussistono rischi per persone o cose in caso di malfunzionamento.

***Backup user program to Flash after download*** (*Esecuzione del backup del programma utente su memoria flash dopo il trasferimento*)

Esegue automaticamente una copia del programma nella memoria Flash<sup>1</sup> di backup. Se questa opzione non è stata selezionata, è comunque possibile eseguire una copia dopo il trasferimento usando la voce di menu *Online, Flash Backup/Restore (In Linea, Copia/Ripristina su/da Flash)*.

***Warn if CPU contains program with different name*** (*Avvertimento in caso la CPU contenga un programma di nome diverso*)

Confronta il nome del programma presente nel PCD con il nome del programma in corso di modifica. Se tali nomi differiscono, verrà visualizzato un messaggio di avvertimento per evitare il trasferimento del programma sulla CPU sbagliata.

***Warn if a running program will be stopped*** (*Avvertimento in caso di arresto del programma in esecuzione*)

Durante l'operazione di trasferimento il PCD si potrebbe arrestare. Selezionando questa opzione si determina la visualizzazione di un messaggio di avvertimento prima che venga arrestato il PCD.

1) PCD2.M170, PCD2.M480, PCD4.M170 e PCD3

***Do not clear Outputs on download or restart*** (Nessuna disattivazione delle uscite durante le fasi di trasferimento o ripartenza)

Questa opzione può essere utile nelle applicazioni HEAVAC. In particolare, impedisce la disattivazione delle funzioni di ventilazione o illuminazione durante il trasferimento di un programma. Non dovrebbe essere utilizzata per altre applicazioni.

***Auto close Up/Download dialog boxes on success*** (Chiusura automatica finestre di dialogo Up/Download (Scaricamento/Caricamento) in caso di trasferimento corretto)

Selezionando questa opzione, le finestre di dialogo Up/Download (Scaricamento/Caricamento) continueranno ad essere visualizzate solo in caso di errore.

### **2.6.2 Caricamento programma su memoria di backup (scheda Flash)**

Se il PCD è equipaggiato con una scheda Flash<sup>1</sup>, la voce di menu *Online, Flash Backup/Restore (Copia/Ripristina su/da Flash)* permette di copiare sulla scheda flash il programma appena caricato nella memoria RAM del PCD, e viceversa. Questa funzione può essere gestita automaticamente selezionando l'opzione di trasferimento appropriata.

### **2.6.3 Backup della memoria e trasferimento del programma applicativo**

E' possibile usare la memoria di backup per trasferire un programma applicativo tra PCD dello stesso tipo:

- Caricare il programma sulla memoria di backup.
- Sezionare l'alimentazione del PCD prima di rimuovere la memoria di backup.
- Scollegare l'alimentazione del PCD prima di inserire la memoria di backup.
- Rimuovere la batteria tampone o premere il pulsante presente sulla memoria di backup per 3 secondi (solo PCD7.R400)
- Ricollegare l'alimentazione al PCD. Durante il ripristino del programma applicativo dalla memoria di backup, i LED lampeggeranno.
- Inserire il modulo batteria per evitare la generazione di un messaggio di errore *battery fail (malfunzionamento batteria)*.



## 2.7 Finestra di Visualizzazione (View)

Questa finestra visualizza delle informazioni solo se il processo di costruzione del programma si conclude correttamente.

### 2.7.1 Struttura dei blocchi organizzativi

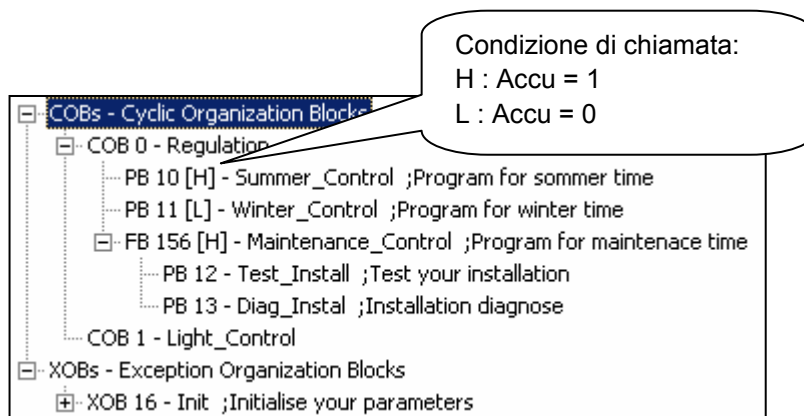
Il programma SAIA®PCD è rappresentato da una struttura di diversi blocchi organizzativi in cui l'utente memorizza i programmi richiesti per l'applicazione.

Ciascun blocco offre un particolare servizio: programmazione ciclica (COB), programmazione sequenziale (SB) sotto programmi (PB), funzioni con parametri (FB), routine eccezionali (XOB).

Dopo la costruzione del programma, il pulsante *Block Structure view (Visualizza struttura blocchi)*, o il comando omonimo del menu *View*, permettono di visualizzare la struttura globale dei blocchi organizzativi che costituiscono il programma.

L'esempio che segue rappresenta un programma costituito da blocchi: COB 0, COB 1, XOB16 PB 10, PB11 e FB 156.

Notare che il COB 0 chiama in modo condizionale tre sotto-blocchi (PB 10, 11 e FB 156). La condizione di chiamata è indicata tra parentesi quadra.



### 2.7.2 Elenco dei blocchi organizzativi

Il pulsante *Block List view (Visualizza elenco blocchi)*, o il comando di menu *Block List (Elenco Blocchi)*, visualizza l'elenco di tutti i blocchi che costituiscono il programma.

  
Visualizza  
Elenco  
Blocchi

Icona	Nome Blocco	Tipo	Numero	Descrizione
☐	Regulation	COB	0	
☐	Summer_Control	PB	10	Program for sommer time
☐	Winter_Control	PB	11	Program for winter time
☐	Maintenance_Control	FB	156	Program for mainten...
☐	Light_Control	COB	1	
☐	Init	XOB	16	Initialise your parameters
☐	Test_Install	PB	12	Test your installation
☐	Diag_Instal	PB	13	Installation diagnose

### 2.7.3 Elenco dei Simboli



Simboli  
Globali



Visualizza  
Elenco

I comandi *Global Symbols (Simboli Globali)* e *View List (Visualizza Elenco)* del menu *View (Visualizza)* permettono di visualizzare i simboli usati dal programma:

- *Global Symbols (Simboli Globali)* visualizza L'Editore dei Simboli che definisce i simboli condivisi da tutti i file della CPU attiva. Tali simboli possono essere modificati all'interno dell'elenco.
- *View List (Visualizza Elenco)* visualizza tutti i simboli utilizzati dalla CPU attiva. Questo elenco non è modificabile. I simboli non utilizzati non vengono visualizzati in questo elenco.

Symbol ▲	Type	Address/...	Scope	Module	Comment
DailyTimer	D	32		Daily Timer.fbd	Daily Timer
HMS	R	2003	AUTO	Daily Timer.fbd	PCD Clock with current time
OFFTIME	R	2004	AUTO	Daily Timer.fbd	Switch off time
ONTIME	R	2005	AUTO	Daily Timer.fbd	Switch on time

### 2.7.4 Riferimenti Incrociati (*Cross-Reference*)

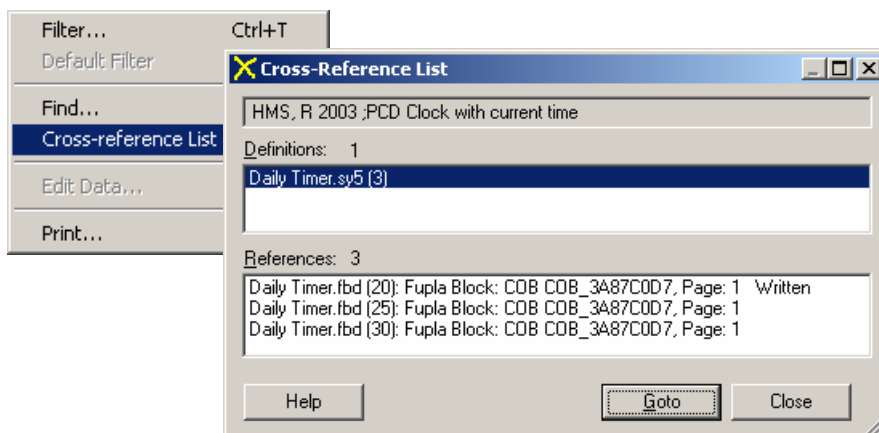
I comandi *Global Symbols* e *View List* offrono la possibilità di selezionare un simbolo e visualizzare l'elenco dei riferimenti incrociati, cioè l'elenco di tutte le locazioni di programma in cui tale simbolo viene utilizzato.

Ogni riga dell'elenco indica il nome del file e il blocco in cui è utilizzato il simbolo selezionato, con l'aggiunta del numero di riga o di pagina. Inoltre, con il termine *Written (Scritto)* viene indicato se il simbolo in quella posizione è stato modificato.

L'elenco *Definitions (Definizioni)* indica dove è stato definito il simbolo, cioè dove può essere trovata la relativa istruzione in Lista Istruzioni EQU. L'elenco *References (Riferimenti)* indica i punti del programma in cui viene utilizzato il simbolo.

Per quanto riguarda i blocchi, la notazione '>>' indica dove può essere trovato il blocco stesso.

Per visualizzare il programma in cui viene utilizzato il simbolo, selezionare la definizione o il riferimento desiderati e premere il pulsante *Goto (vai a)*.



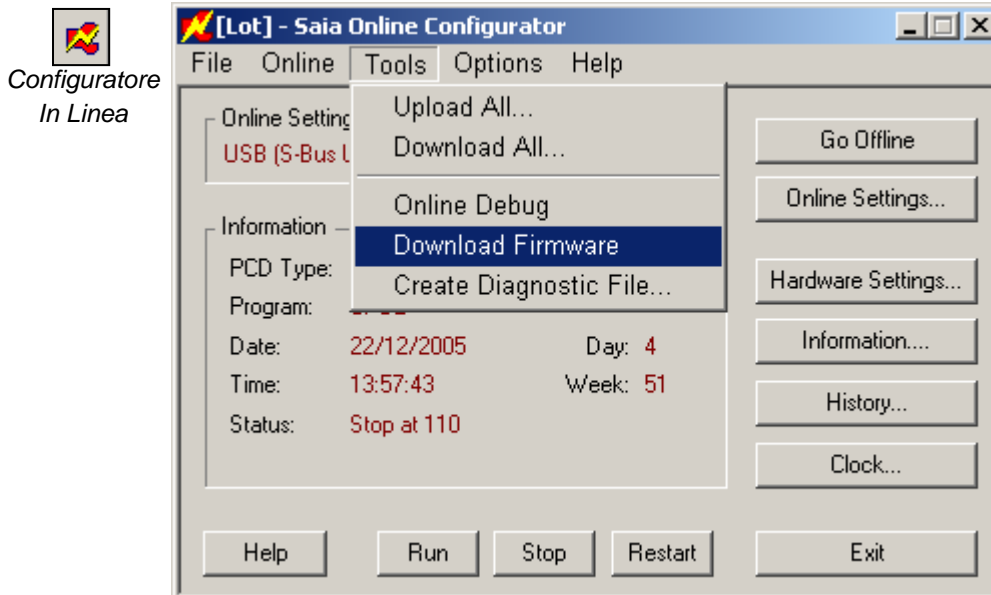
## 2.8 Backup del Programma

Il risultato di una qualsiasi modifica apportata ad un programma PCD, talvolta può essere incerto. Ad esempio, si potrebbe non essere certi che i file sorgenti disponibili siano quelli corrispondenti all'ultima versione, oppure si potrebbe non avere una conoscenza completa, ecc.

Per evitare possibili problemi dovuti a queste incertezze è possibile salvare l'intero contenuto della memoria del PCD e ripristinarlo all'accensione.

Il comando *Upload All (Salva Tutto)* nel menu *Tools* del Configuratore In linea, permette di salvare l'intera memoria del PCD in un unico file (comprendente il programma, le impostazioni hardware, i valori dei registri, flag, contatori, DB e testi).

Per ripristinare il programma nella memoria del PCD, utilizzare il comando *Download All (Trasferisci Tutto)* dal menu *Tools (Strumenti)*, e selezionare il file contenente i file di backup.



Nota:

I backup possono essere eseguiti solo se è presente una memoria di backup PCD7.R400/R500

## 2.9 File a trasferimento autonomo

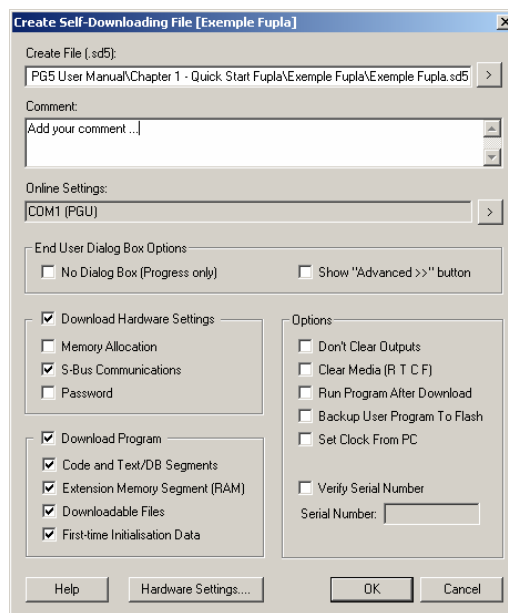
La funzione di generazione di file a trasferimento autonomo semplifica il trasferimento di impostazioni hardware e programmi ai PCD operanti sul campo.

Questo tool PG5 permette infatti di predisporre un file *'sd5'* contenente tutte le informazioni necessarie per aggiornare i programmi e le configurazioni del PCD. Il programmatore PG5 semplicemente invierà poi via e-mail questo file alla persona incaricata della gestione del PCD sul campo.

All'apertura del file *'sd5'* file, verrà visualizzata la finestra di dialogo dedicata al trasferimento dati. Alcuni parametri ed opzioni corrisponderanno a quelle predefinite nel progetto PG5. L'addetto presente sul campo può sia lasciare tali opzioni invariate o modificarle in base alle esigenze prima del trasferimento dati al PCD.

Ciò significa che non è richiesta alcuna particolare conoscenza dell'ambiente PG5 per procedere al trasferimento di programmi o impostazioni hardware al PCD. Inoltre, questo tool permette di trasferire programmi e impostazioni hardware senza dover installare sul campo il pacchetto PG5 e/o dover acquisire una licenza d'uso. In ogni caso, è necessario installare sul campo il pacchetto *Stand Alone Online Tools (Tool In Linea Stand-Alone)*.

### 2.9.1 Preparazione di un file '.sd5'



Il file viene predisposto in base alle informazioni contenute nella CPU attiva, secondo quanto visualizzato nella finestra dell'Albero del Progetto. E' consigliabile controllare che le Impostazioni In Linea (Online Settings) e le Impostazioni Hardware (Hardware Settings) siano correttamente configurate nonché eseguire una "costruzione" del programma CPU prima di generare il file '.sd5' file.

La voce di menu *CPU, Create Self-Downloading File (CPU, Crea file a Trasferimento Autonomo)* permette di configurare i parametri e le opzioni desiderati per il trasferimento autonomo sul campo.

I parametri e le opzioni disponibili sono identici a quelli selezionabili mediante i menu *Online, Hardware Settings, Download (In Linea, Impostazioni Hardware, Trasferimento)* e *Online, Download Program ... (In Linea, Trasferimento Programma...)*. Tuttavia, sono presenti alcuni parametri supplementari:

#### **Create Files (\*.sd5) (Crea File (\*.sd5))**

Permette di specificare il nome ed il percorso del file .sd5.

#### **Show "Advanced >>" button (Mostra pulsante "Avanzate>>")**

Permette di nascondere tutte le preselezioni presenti in questa finestra di dialogo durante il trasferimento autonomo

#### **No Dialog box (Progress only) (Nessuna Finestra di Dialogo (solo avanzamento))**

Trasferisce il file '.sd5' senza visualizzare alcuna finestra di dialogo (modo "silenzioso")

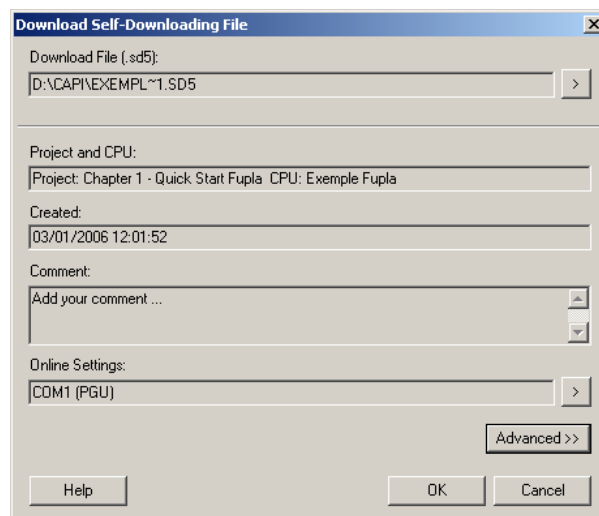
**Verify Serial Number** (*Verifica Numero di Serie*)

La funzione di trasferimento autonomo verifica che il numero di serie del PCD corrisponda a quello specificato nel campo *Serial Number (Numero di Serie)*. Questo numero di serie è univoco per ciascun PCD e può essere pertanto utilizzato per accertarsi che il trasferimento venga eseguito sul PCD desiderato.

**Nota:**

Il numero di serie è supportato solo dai nuovi sistemi PCD3. Per leggere il numero di serie In Linea, è possibile usare il Configuratore In Linea, selezionando la voce di menu *Online, Information (In Linea, Informazioni)*.

## 2.9.2 Trasferimento di in file '.sd5'



Per trasferire su PCD il contenuto di un file '.sd5', controllare innanzitutto che sul PC utilizzato sia installato il pacchetto PG5 o il pacchetto *Stand Alone Online Tools (Tool In Linea Stand-Alone)*. Consultare la guida all'installazione di PG5 per maggiori dettagli.

Aprire il file '.sd5' mediante la *Gestione Risorse* dell'ambiente *Windows*, facendo doppio clic con il mouse sul file stesso. Verrà visualizzata una finestra di dialogo simile a quella sopra riportata.

E' possibile riconfigurare le opzioni di trasferimento dei contenuti del file facendo clic sul pulsante *Advanced (Avanzate)*. Il trasferimento può essere avviato facendo clic sul pulsante *OK*.



## Indice

---

3	Risorse PCD	3-3
3.1	Introduzione	3-3
3.2	Risorse Hardware	3-4
3.2.1	Ingressi e Uscite Digitali	3-4
3.2.2	Orologio	3-5
3.2.3	Ingressi di Interrupt	3-6
3.3	Risorse Interne (Software)	3-7
3.3.1	Flag	3-7
3.3.2	Registri	3-8
3.3.3	Costanti	3-9
3.3.4	Temporizzatori e Contatori	3-10
3.3.5	Testi & Data Block	3-13
3.3.6	Tabella Riassuntiva	3-15
3.4	Editore dei Simboli (Symbol Editor)	3-16
3.4.1	Elementi di una Risorsa	3-16
3.4.2	Raggruppamento dei Simboli	3-17
3.4.3	Ambito dei Simboli	3-17
3.4.4	Simboli Locali	3-18
3.4.5	Simboli Globali	3-18
3.4.6	Definizione di un Simbolo Globale	3-19
3.4.7	Definizione dei simboli utilizzati per le reti di comunicazione	3-19
3.5	Come lavorare con i simboli	3-20
3.5.1	Scrittura di un Elenco di Simboli	3-20
3.5.2	Aggiunta di più simboli all'Editore dei Simboli	3-21
3.5.3	Simboli di Riferimento	3-22
3.5.4	Come importare dei simboli da istruzioni "EQUATE"	3-23
3.5.5	Come importare dei simboli da un'altra applicazione	3-23
3.5.6	Aggiunta di simboli durante la scrittura del programma in Lista Istruzioni	3-23
3.5.7	Aggiunta di simboli durante la scrittura del programma in FUPLA	3-24
3.5.8	Trasferimento di simboli	3-25
3.5.9	Completamento Automatico dei Simboli	3-26
3.5.10	Allocazione Automatica	3-26
3.5.11	Inserimento di testi	3-27
3.5.12	Inserimento di DB	3-28
3.5.13	Ricerca di un Simbolo	3-28
3.5.14	Disposizione dei Simboli	3-29
3.5.15	Riorganizzazione in "List View" (Disposizione a Lista)	3-30
3.5.16	Esportazione di simboli	3-31
3.5.17	Importazione di simboli	3-33
3.5.18	Inizializzazione dei simboli	3-35
3.5.19	Nomi dei Simboli	3-36
3.5.20	Parole Riservate	3-36





## 3 Risorse PCD

---

### 3.1 Introduzione

Questo capitolo fornisce una panoramica generale dei tipi di dati che potranno essere utilizzati nella scrittura dei programmi applicativi.

Le prime due sezioni riassumono tutti gli elementi comunemente utilizzati nei SAIA®PCD quali Ingressi/Uscite o Flag, con i relativi campi di indirizzi e modalità di utilizzo.

Le ultime due sezioni indicano come utilizzare questi elementi all'interno dell'Editore dei Simboli (Symbol Editor).

## 3.2 Risorse Hardware

Ogni programma è costituito da funzioni, che permettono all'utente di leggere, scrivere e manipolare tipi diversi di risorse. Le risorse che ci permettono di interagire con l'applicazione sono denominate Risorse Hardware.

### 3.2.1 Ingressi e Uscite Digitali

I/O 1 bit (0/1) di informazione

	numero massimo di I/O
PCD1	32 ( 64 <sup>3)</sup> )
PCD2.M120/M150	64/96/128 (255 <sup>3)</sup> ) <sup>1)</sup>
PCD2.M170 + PCD3.C100	510 <sup>2)</sup> <sup>3)</sup>
PCD2.M480 + PCD3.C100/PCD3.M	1023 <sup>1)</sup> <sup>3)</sup>
PCD3.Mxxxx	1023 <sup>1)</sup> <sup>3)</sup>
PCD4	510 <sup>2)</sup>
PCD6	5100 <sup>2)</sup>

- 1) Gli indirizzi 255 (e 511 per PCD2.M170) sono riservati alla funzione watchdog
- 2) Gli indirizzi 255, 511, 767, 1023, ... , fino a 5119) sono riservati alla funzione watchdog
- 3) con schede di ingresso PCD2/3.E160 e/o schede di uscita PCD2/3.A460

Gli ingressi e le uscite rappresentano i segnali diretti a, o provenienti dal PCD. Gli ingressi indicano lo stato dei fine corsa, pulsanti, rilevatori di prossimità, sensori, ecc. Le uscite permettono invece di attivare valvole, lampade, motori C/A, ecc.

Le uscite possono essere lette e scritte. Gli ingressi possono solo essere letti.

Gli ingressi e le uscite possono essere aggiunti al PCD inserendo delle schede di I/O negli appositi slot del PCD. L'indirizzo iniziale di uno slot è definito direttamente dalla sua posizione (PCD1/2/3 e 4) oppure tramite switch (PCD 6).

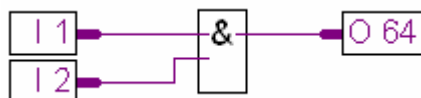
L'esempio che segue attiva l'uscita O 64 solo se gli ingressi I1 e I2 sono entrambi a livello alto.

Un altro modo per rappresentare funzioni di questo tipo è quello di utilizzare le equazioni booleane:  $O\ 64 = I\ 1 * I\ 2$

#### Programma in Lista Istruzioni:

```
COB  0
      0
STH  I 1
ANH  I 2
OUT  O 64
ECOB
```

#### Programma FUPLA:



Fbox: *Binary, And (Binario, AND)*

### 3.2.2 Orologio

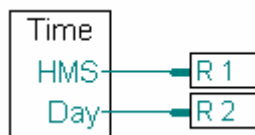
La maggior parte dei PCD (PCD1.M120/130 e tutti i PCD2/3/4/6) hanno un orologio in tempo reale incorporato (RTC – real time clock ). Data e ora possono essere caricate in un registro con una istruzione speciale.

L'esempio che segue indica come effettuare la lettura dell'orologio in un programma.

#### Programma in Lista Istruzioni:

```
COB 0
      0
RTIME R 1
ECOB
```

#### Programma FUPLA:



Fbox: *Time Related, Read time*  
(Data/Ora, Leggi Ora)

Questo programma legge l'ora dall'orologio e ne copia il valore all'interno del registro R1. L'ora è rappresentata nel seguente modo:

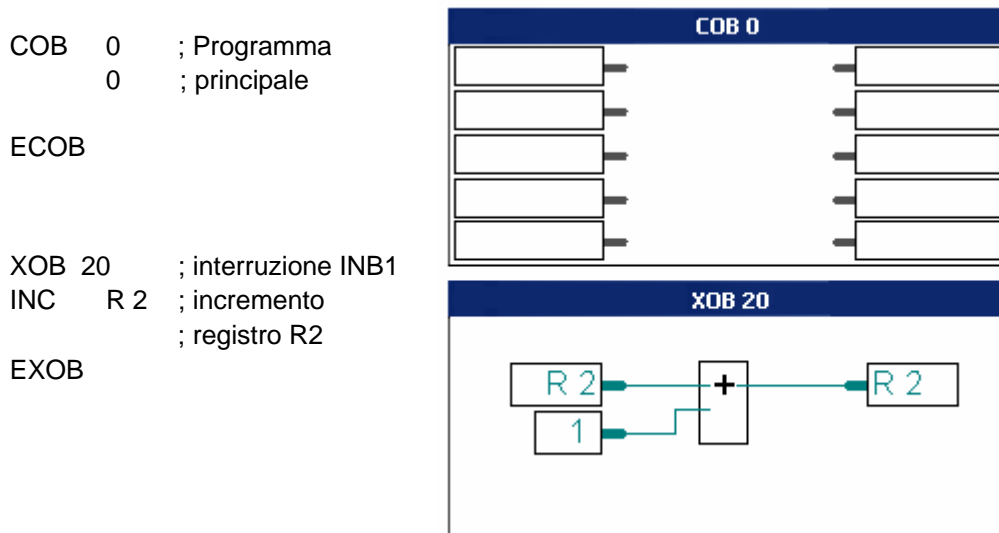
```
R 1 = 093510      ore 09, 35 minuti e 10 secondi
R 2 = 073030210  settimana 07, giorno n° 3 ( Mercoledì ), 10 Febbraio 2003.
```

### 3.2.3 Ingressi di Interrupt

Alcuni PCD<sup>1</sup> hanno due ingressi di interrupt denominati INB1 e INB2<sup>2</sup>. Quando su uno di questi ingressi si ha un fronte positivo, il normale ciclo di programma verrà interrotto e il PCD eseguirà uno speciale blocco di programma denominato XOB20 o XOB25 ( XOB20 per INB1 e XOB25 per INB2 ).

La frequenza massima su questi ingressi è di 1000 interrogazioni al secondo.

L'esempio seguente dimostra come conteggiare gli impulsi provenienti da INB1.



**Programma in Lista Istruzioni:**

**Programma FUPLA:**

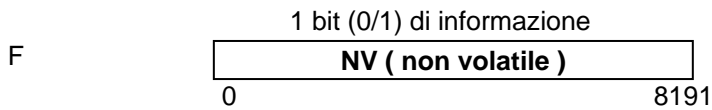
- 1) PCD1.M120/130, PCD2.M120/150, PCD2/4.M170, PCD2.M480 (4 ingressi di Interrupt IN0 ... IN3), PCD3.M e PCD6.M3
- 2) Per ulteriori informazioni fare riferimento ai manuali hardware PCD.



Le limitazioni imposte dal filtro di ingresso (utilizzato per proteggere i normali ingressi digitali contro i disturbi ed i rimbalzi dei contatti meccanici) impediscono all'ingresso digitale di conteggiare gli impulsi con frequenza maggiore di 50 Hz. Gli ingressi di interrupt rappresentano quindi una soluzione alternativa interessante per questo tipo di applicazione. Essi evitano la necessità di utilizzare le schede di conteggio PCD2.H1 o PCD4.H1, che hanno una frequenza di conteggio massima variabile da 10 a 160 kHz, in base al tipo di modulo.

### 3.3 Risorse Interne (Software)

#### 3.3.1 Flag



Una flag memorizza un'informazione costituita da un solo bit. Sono disponibili 8192 flag (è valida anche la flag 0). Per default, le flag sono non volatili, questo significa che se si spegne il PCD, e la flag è a 1, alla riaccensione del PCD la flag sarà ancora a 1 (supponendo che la batteria sia carica). Le flag volatili verranno invece riportate tutte al valore 0 allo spegnimento del PCD. L'eventuale configurazione di una o più flag come "volatili" può essere configurata in Software Setting (*Impostazioni Software*). Questo verrà spiegato più avanti. L'esempio che segue scrive un livello alto (1) nella Flag numero 11 quando uno dei due ingressi (1 o 3) è alto. Equazione Booleana:  $F_{11} = I_1 + I_3$

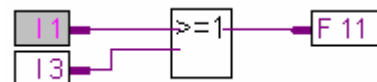
#### Come utilizzare le flag in un programma

Programma in Lista Istruzioni:

```

COB  0
      0
STH  I 1
ORH  I 3
OUT  F 11
ECOB
    
```

Programma in FUPLA:



Fbox : Binary, Or

Per default le flag sono non volatili. Se si desidera trasformarle in volatili, sarà necessario specificare questa condizione nelle impostazioni software (*Software Settings*). (Vedere l'esempio seguente).

#### Configurazione delle Flag

Doppio clic con il mouse

**Software Settings [Main CPU]**

Registers/Texts/DBs | Timer/Counters | **Flags** | Directories | Advanced

Volatile Flags Set Defaults

Last Volatile Flag:  Flags up to this address are volatile and are set to zero at start-up.

Dynamic Space			
	First Address	Last Address	
Volatile Flags:	2500	3000	0 / 501
Non-Volatile Flags:	7500	8191	2 / 690

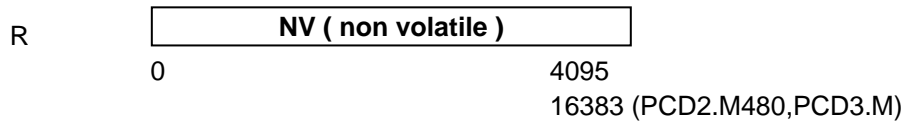
Save As... | OK | Cancel | Help

### 3.3.2 Registri

Valore su 32 bit

Integer: -2 147 483 648 ÷ +2 147 483 647

Virgola mobile: -9.22337E+18 ÷ +9.22337E+18



Un registro può contenere valori interi o in virgola mobile. I registri sono estremamente utili per le operazioni aritmetiche o per le operazioni con valori analogici, tipici delle attività di misura e regolazione. Si possono avere fino a 4096 Registri. I registri sono risorse di tipo non volatile. In FUPLA, le linee collegate ad un registro assumono colori diversi in base al contenuto del registro stesso. Se il registro contiene un valore in virgola mobile, le linee sono di colore giallo, se invece contiene un valore intero, sono di colore verde. Non è possibile gestire l'interazione tra un valore intero e un valore in virgola mobile. Ad esempio non è possibile sommare un valore intero ad uno in virgola mobile. Prima di effettuare la somma, uno dei due valori dovrà quindi essere convertito.

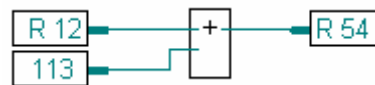
#### Come utilizzare i registri nei programmi

L'esempio seguente somma il numero 113 al contenuto del registro 12 e carica il risultato nel registro 54:  $R\ 54 = R\ 12 + 113$

**Programma in Lista Istruzioni:**

```
COB 0
    0
ADD  R 12
    K 113
    R 54
```

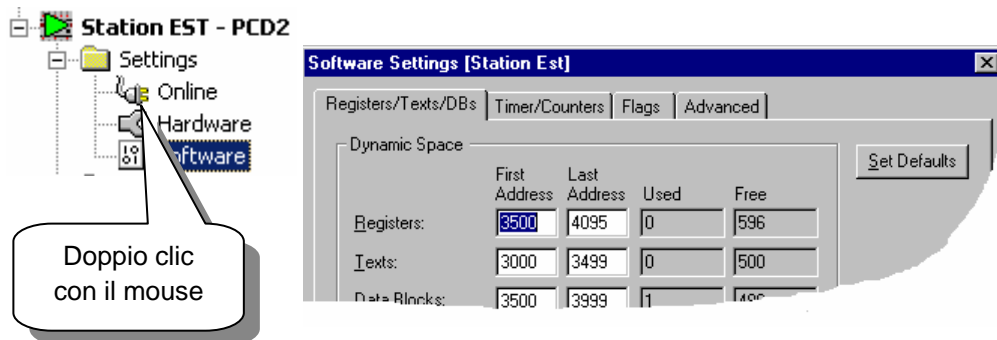
**Programma in FUPLA:**



Fbox: *Entier, Addition*

#### Configurazione dei Registri

L'allocazione dinamica delle risorse è una potente funzionalità che è stata introdotta per svincolare l'utente dall'obbligo di dover specificare un indirizzo fisso per ogni risorsa richiesta. Le risorse dinamiche possono essere utilizzate definendo un nome simbolico per la risorsa senza specificare un indirizzo. Non sarà necessario cambiare queste impostazioni finché non sorgerà l'esigenza di scrivere programmi di grosse dimensioni con un elevato numero di registri.



In caso di errori del tipo *Auto allocation overflow for type: R (Superamento dello spazio ad allocazione automatica per il tipo: R)* sarà necessario ampliare la configurazione dello spazio dinamico.

### 3.3.3 Costanti

Valore su 32 bit

Intero: -2 147 483 648 ÷ +2 147 483 647

Virgola mobile: -9.22337E+18 ÷ +9.22337E+18

Le costanti sono dei valori fissi che non cambiano durante l'esecuzione del programma. Le costanti vengono scritte in un registro.

**Esempio:** coefficiente fisso tipo.  $\pi$  (PI) = 3,1415.

L'esempio seguente carica un valore fisso ( 100 ) nel registro R4. Il registro R4 viene quindi diviso per 0.25. Poiché il registro R4 contiene un valore intero che deve essere diviso per un valore decimale ( 0.25 ), sarà necessario convertire R4 in un valore decimale. In questo caso il registro R4 viene copiato in R35 (registro sicuramente non utilizzato), R35 viene convertito in un valore decimale, e quindi diviso per 0.25. Il risultato della divisione viene caricato in R5. R5 viene quindi copiato in R6, e R6 viene convertito in un valore intero.

#### Come utilizzare le costanti nel programma

##### Programma in Lista Istruzioni:

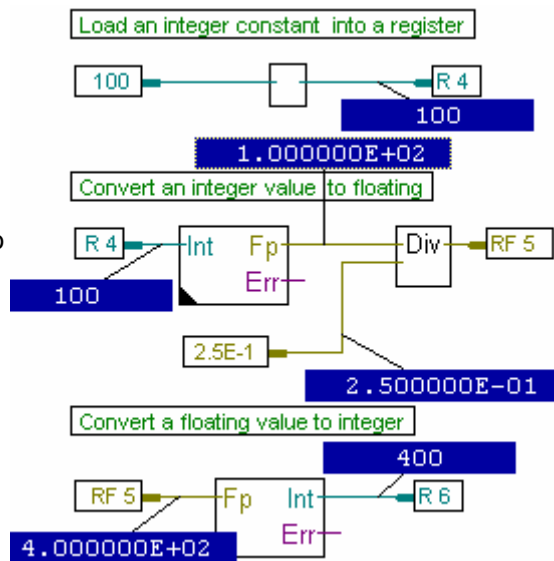
```

COB 0 ; Blocco ad
      0 ; organizzazione ciclica
LD R 4 ; carica 100 in R4
  100

COPY R 4 ; converti il valore intero
     R 35 ; in virgola
IFP R 35 ; mobile
     0
LD R 36 ; Carica 0,25 in R 36.
  2.5e-1
FDIV R 35 ; dividi il valore per
      0.25
     R 36
     R 5 ; e carica il risultato
          ; in R5

COPY R 5 ; converti nuovamente il
     R 6 ; risultato in valore
          ; intero
FPI R 6
     0
    
```

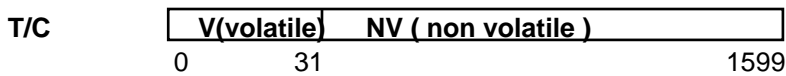
##### Programma FUPLA:



- Fbox:
- Integer, Move (Intero, Sposta)
  - Converter, Int to float (Convertitore, da Intero a Virgola Mobile)
  - Floating point, Divide (Virgola Mobile, Dividi)
  - Converter, Float to Int (Convertitore, da Virgola Mobile a Intero)



### 3.3.4 Temporizzatori e Contatori

Valore su 31 bit ( 0 ... 2 147 483 648)



I temporizzatori e i contatori possono assumere un valore compreso tra 0 e 2 147 483 648 (31Bit) e condividono lo stesso campo di indirizzi: da 0 a 1599. Normalmente gli indirizzi da 0 a 31 sono dedicati ai temporizzatori (timer), mentre gli indirizzi da 32 a 1599 sono dedicati ai contatori.

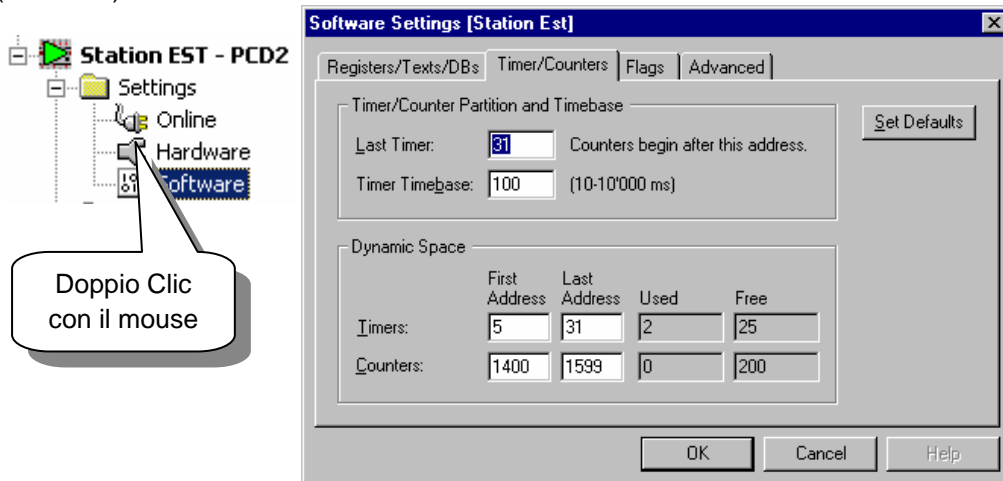
Ovviamente, la configurazione può essere effettuata in base alle esigenze specifiche. I temporizzatori hanno per default una base dei tempi di 100ms. (Questo significa che il sistema decreterà di 1 ciascun temporizzatore, ogni 100ms). La base dei tempi può essere cambiata nella finestra di dialogo *Software Settings (Impostazioni Software)*, insieme alla configurazione degli indirizzi dei Temporizzatori / Contatori. I Temporizzatori sono di tipo volatile, mentre i Contatori sono di tipo non volatile.

Temporizzatori e Contatori possono contenere esclusivamente valori positivi. Tali valori possono essere cambiati caricando un nuovo valore con l'istruzione LD. I valori dei Temporizzatori possono solo essere decrementati. I Contatori possono invece contare avanti o indietro utilizzando l'istruzione INC/DEC. (Inc  Dec .

I Temporizzatori e i Contatori possono anche essere utilizzati con le istruzioni binarie. Quando un Contatore o un Temporizzatore contiene un valore diverso da 0 il suo stato è Alto (1), quando contiene il valore 0 il suo stato è Basso (0).

#### Configurazione dei Temporizzatori / Contatori

La ripartizione del campo di indirizzi tra temporizzatori e contatori può essere modificata nella finestra di dialogo *Software Settings (Impostazioni Software)*. Nella stessa finestra è anche possibile variare la temporizzazione base di 100ms (timebase).



#### Informazioni Tecniche

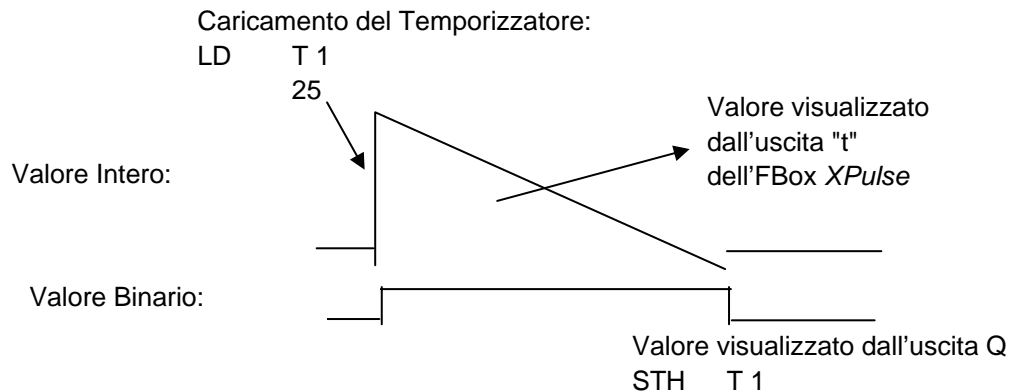
Maggiore è il numero di temporizzatori dichiarate, maggiore sarà il carico sulla CPU. Lo stesso effetto si ottiene se si diminuisce il valore della base dei tempi. Tenere in considerazione questo fatto prima di cambiare il numero di temporizzatori o ridurre il valore della base dei tempi.

Esempio: 100 temporizzatori impegnano circa il 2% della capacità di CPU.



### Esempio di Temporizzatore

Supponiamo di avere un segnale a livello alto sull'ingresso 4. Sul fronte di salita di questo segnale vogliamo commutare a livello alto anche il segnale dell'uscita 65. Questo segnale dovrà avere una durata di 2,5 secondi.



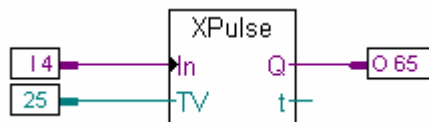
### Soluzione:

#### Programma in Lista Istruzioni:

```

COB  0      ; blocco ad organizzazione ciclica 0
      0      ; valore del timeout
STH  I 4    ; se l'ingresso 4 rileva
DYN  F 12   ; un fronte di salita
LD   T 1    ; carica il timer1
      25    ; a 2,5 secondi
STH  T 1    ; copia lo stato del temporizzatore
OUT  O 65   ; sull'uscita O65
ECOB
    
```

#### Programma in FUPLA:



Fbox : *Time related, Exclusive pulse*  
 (Temporizzatore, Impulso esclusivo)



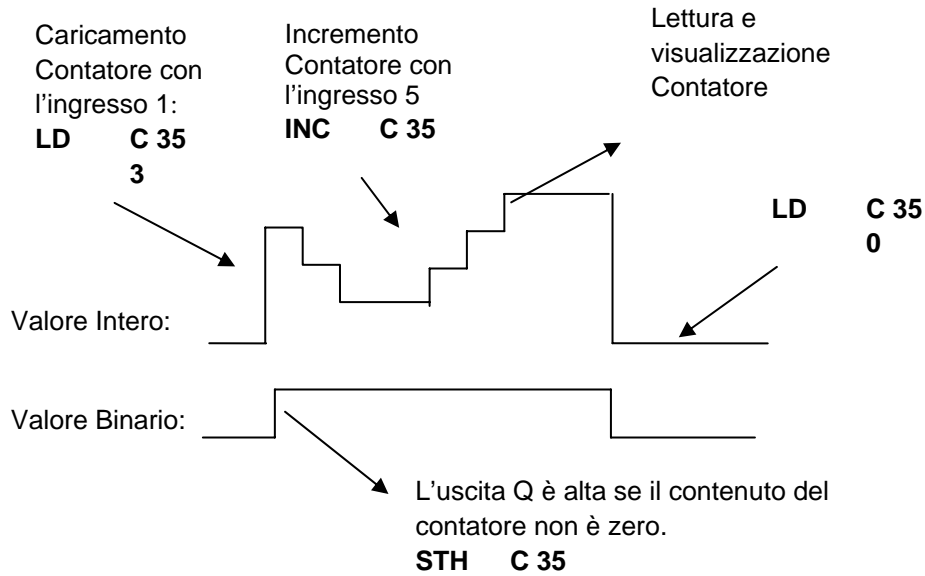
### Informazioni Tecniche

Nei SAIA PCD i temporizzatori vengono decrementati alla frequenza definita nella finestra di dialogo *Software Settings, Timer, Time-base (Impostazioni Software, Temporizzatore, Base Temp)* (normalmente 100ms). Il tempo effettivo, definito da una costante caricata in un temporizzatore varia con il variare del parametro Time-base. Questo significa che se viene variato il parametro Time-base, dovranno essere cambiati anche tutti i valori di caricamento dei Temporizzatori. Per evitare questo problema, si può utilizzare il tipo di dati "Time" per dichiarare il valore di caricamento dei temporizzatori. Se si utilizza un valore "Time", il linker calcola il valore di caricamento effettivo dei temporizzatori in base al parametro Time-base.

BL_3DE393BA	COB		
DelayTime	K Constant	T#100MS	100 milliseconds
OneDay	K Constant	T#3600S	3600 secondes

### Esempio di Contatore

Supponiamo di voler programmare un contatore che si incrementi di 1 quando l'Ingresso 5 riceve un segnale. Il contatore si deve decrementare di 1 quando a ricevere il segnale è l'ingresso 6. (Il conteggio deve essere attivato sul fronte di salita del segnale di Ingresso). Il contatore potrà essere azzerato inviando un segnale a livello alto sull'ingresso 2. Il contatore dovrà essere precaricato con il valore 3.



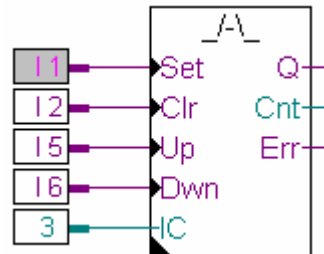
### Soluzione:

#### Programma in Lista Istruzioni:

```

COB 0 ; blocco ad organizzazione
      0 ; ciclica
STH I 1 ; Se l'ingresso 1 è uguale a 1
LD C 35 ; carica il contatore 35
      3 ; con il valore 3
STH I 2 ; se l'ingresso 2 è uguale a 1
LD C 35 ; carica il contatore
      0 ; con il valore zero
STH I 5 ; Se è presente un fronte di salita
DYN F 13 ; sull'ingresso 5
INC C 35 ; incrementa il contatore 35
STH I 6 ; Se è presente un fronte di salita
DYN F 14 ; sull'ingresso 6
DEC C 35 ; decrementa il contatore
ECOB
    
```

#### Programma FUPLA:



Fbox :  
 Counter, Up down with preset and clear  
 (Contatore, Su-Giù con preset e  
 azzeramento)

### 3.3.5 Testi & Data Block

TEXT/DB	<b>Memoria Principale</b>		<b>NV</b>
	0		3999
	<b>Memoria Estesa</b>		<b>NV</b>
			4000
		(PCD4/6)	7999
		(PCD2)	5999
		(PCD1)	4999
		(PCD2.M480, PCD3.M)	8191

I Testi (stringhe di caratteri) e i DataBlock (DB) non sono volatili. I Testi sono utilizzati per: messaggi sul display, testi da inviare ad un pager, stringhe iniziali per modem e così via. I DB sono invece utilizzati per la registrazione di dati, tabelle e così via.



#### Informazioni Tecniche

##### Dove vengono salvati i Testi / DB?

I registri, flag, temporizzatori e contatori sono gestiti dal sistema e memorizzati in una piccola porzione di RAM, separata dalla memoria principale.

I DB e i Testi invece sono memorizzati nella memoria principale, insieme ai programmi utente. Se come memoria principale si vuole utilizzare una FLASH Eprom o una normale EPROM, è necessario ricordare che in modalità Run, è possibile leggere da questo tipo di memoria ma non scrivervi. Non è quindi possibile modificare il contenuto dei DB (ad esempio i Dati di Login). Nella maggior parte dei casi questo non rappresenta un problema, ma se si vuole poter leggere e **scrivere** il contenuto dei DB, allora sarà necessario memorizzarli nella memoria estesa => a partire dall'indirizzo 4000. (La memoria estesa è infatti sempre di tipo RAM, quindi può essere letta e scritta liberamente).

#### Esempio: Dichiarazione di DB & Testi

TEXT 10 "Buongiorno!" ; Il Testo n° 10 contiene la stringa Buongiorno!

TEXT 11 [7]"Salve" ; Il Testo n° 11 è lungo 7 caratteri di cui gli ultimi  
; 5 contengono la scritta Salve e i primi due  
; contengono degli spazi.

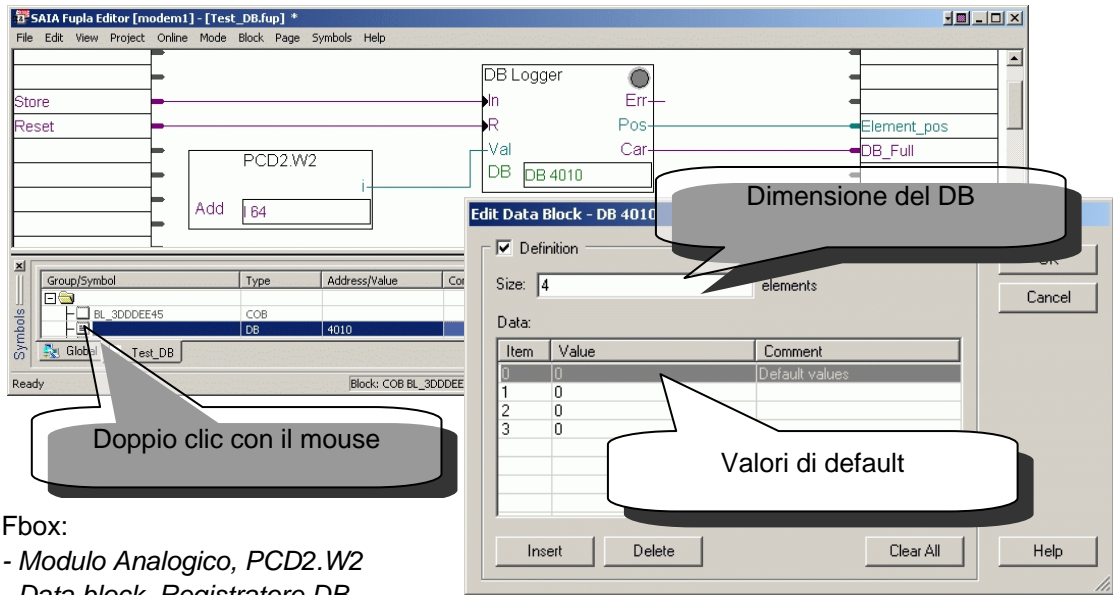
DB 12 45,46,78,999,0 ; DB n° 12 con 5 valori interi: 45,46,78,999,0

DB 13 [10] ; Il DB n° 13 contiene 10 valori inizialmente  
; forzati a 0.

DB 14 [4] 2,3 ; I DB n° 14 contiene 4 valori. I primi due valori  
; sono rispettivamente 2 e 3, mentre i rimanenti  
; sono 0.

### Esempio: Registratore di Dati in FUPLA

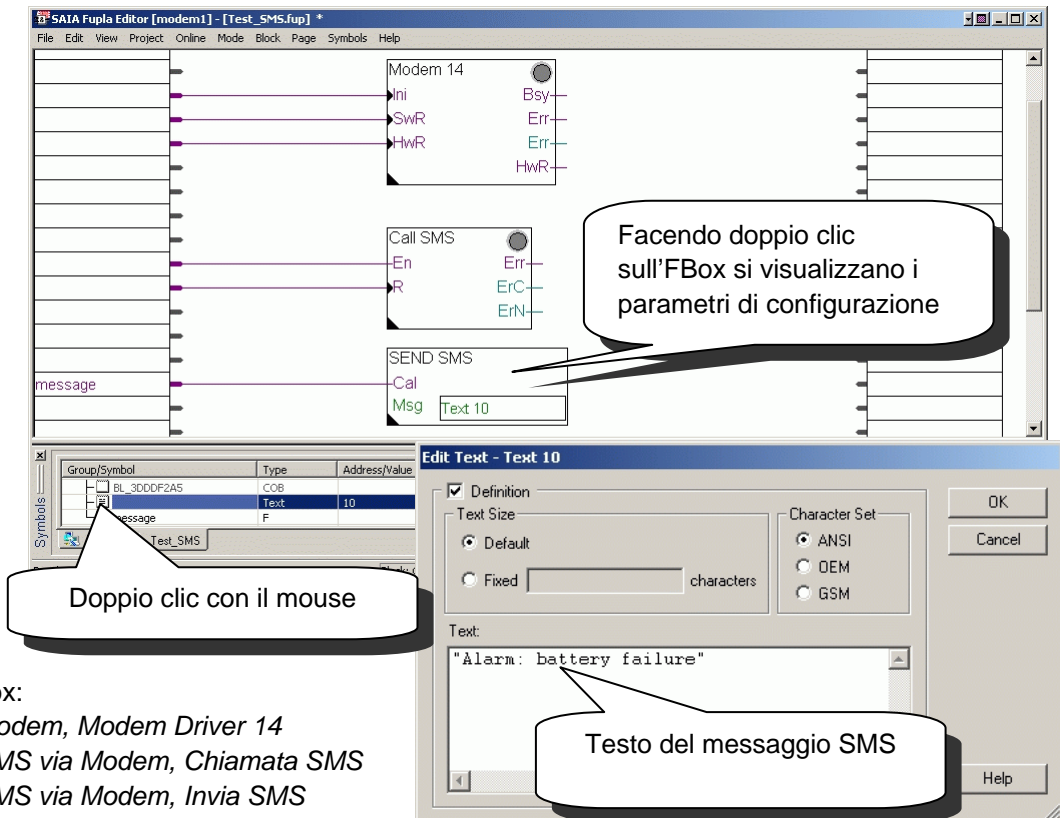
Vediamo ora come si possono registrare facilmente dei valori provenienti da una scheda analogica nel DB 4010. Ad ogni attivazione del segnale "Store", il valore analogico viene letto e successivamente scritto nel DB identificato dal numero 4010.



- Fbox:
- Modulo Analogico, PCD2.W2
  - Data block, Registratore DB

### Esempio: Invio di un SMS in FUPLA

Vediamo ora come inviare un messaggio SMS usando lo stato binario di un ingresso digitale o di una flag. Il messaggio è definito nel Testo 10. Notare i triangolini neri presenti nell'angolo inferiore sinistro degli FBox usati in questo esempio. Essi indicano che alle funzioni interessate è associata una *finestra di configurazione* (*adjust window*) contenente i parametri relativi al pager di destinazione o al numero del modem. Le finestre di configurazione possono essere visualizzate facendo doppio clic con il mouse al centro dell'Fbox interessato.



- Fbox:
- Modem, Modem Driver 14
  - SMS via Modem, Chiamata SMS
  - SMS via Modem, Invia SMS

### 3.3.6 Tabella Riassuntiva

Descrizione	Tipo di Risorsa	Operando	Binario	Numerico	Volatile
Ingressi	I	1) 0...8191	0,1		
Uscite	O	1) 0...8191	0,1		
Flag	F	0...8191	0,1		<sup>2)</sup> No
Temporizzatori	T	2) 0...31	0,1	0 ... 2 147 483 648	Si
Contatori	C	2) 32...1599	0,1	0 ... 2 147 483 648	No
Registri	R	0...4095 5) 0...16383		-2 147 483 648...+2 147 483 647 -9.22337E+18...+9.22337E+18	No
Testo	X	3) 0...3999 4) 4000 ...		Stringa di max. 3072 caratteri	No
Data block	DB	3) 0...3999 4) 4000 ...		Max. 382 valori (accesso lento) Max.16 383 valori (accesso rapido)	No

- 1) dipendente dal PLC e dalla relativa configurazione di ingressi, uscite
- 2) per default, configurabile mediante il menu *Software Setting (Impostazioni Software)*
- 3) memorizzato nella stessa memoria usata per il salvataggio dei programmi (RAM / EPROM / FLASH)
- 4) memorizzato nella memoria estesa (RAM)
- 5) PCD2.M480, PCD3.M

### 3.4 Editore dei Simboli (Symbol Editor)

Prima di iniziare a programmare è necessario elencare tutti gli elementi che prevediamo di utilizzare (numero di ingressi o di uscite, numero di temporizzatori, e così via).

Tutti questi elementi devono essere conosciuti dal PG5. Questo risulterà estremamente utile per poter individuare gli elementi all'interno dei file di programma, indicarci dove abbiamo commesso un errore di programmazione, o aiutarci durante il processo di debugging. Per questo motivo tutti gli elementi che prevediamo di utilizzare verranno elencati in uno strumento centrale denominato *Editore dei Simboli (Symbol Editor)*.

Utilizziamo l'espressione "Simbolo" anziché elemento in quanto vogliamo enfatizzare il fatto che ogni elemento è caratterizzato da un nome (Simbolo). Inoltre l'assegnazione di un nome a tutte le risorse facilita la lettura del programma.

#### 3.4.1 Elementi di una Risorsa

The screenshot shows the Symbol Editor window with a table of symbols. The table has columns for Group, Symbol, Type, Address/Value, and Comment. The following table represents the data shown in the screenshot:

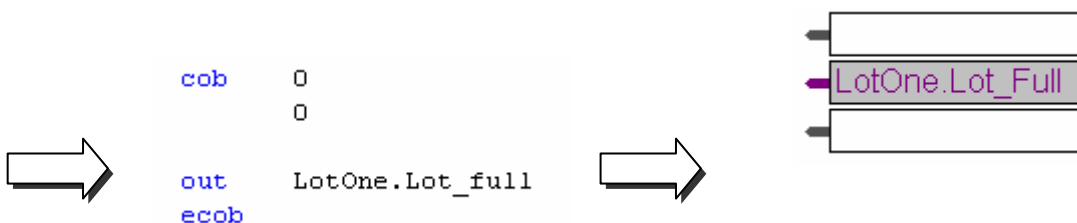
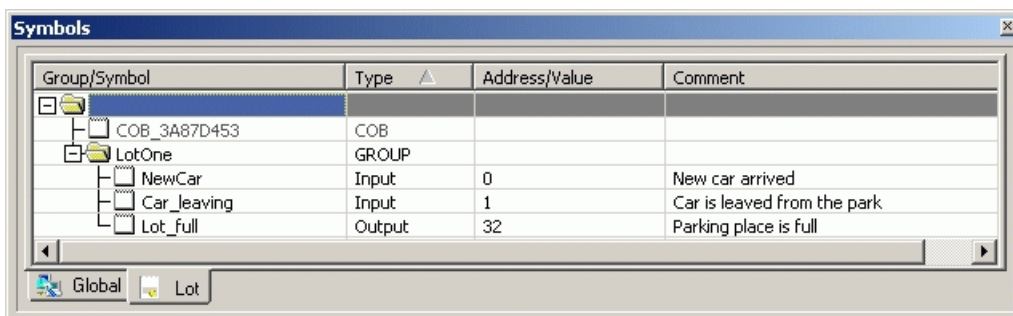
Group	Symbol	Type	Address/Value	Comment
	NormalRun	I	3	Machine is in normal run
	Cond_Run	I	4	Machine is in conditional...
	OilHigh	I	5	Oil Level is too high
	Emergency	I	7	Emergency stop on the ...
	IntermediatFlag	F		
	OilPump	O	20	Oilpump
	OilPumpProg	COB	3	

The callouts provide the following information:

- Nome di una risorsa:** (fino a un massimo di 80 caratteri)
- Tipo del simbolo:** Qui viene specificato il tipo di risorsa che si sta utilizzando. Ad esempio Ingresso o Registro...
- Commento:** Aggiungere un commento esaustivo ad ogni risorsa. Questo faciliterà la lettura del programma.
- Globali:** i simboli presenti in questo elenco sono resi noti a tutti i file della CPU.
- Nome File** a cui appartengono i simboli. I simboli non verranno resi noti agli altri file.
- Indirizzo/Valore:** È necessario comunicare al PG5 il numero di ingresso o di uscita da assegnare a questo simbolo. In caso di risorse interne (qualsiasi elemento diverso da un ingresso o da una uscita) non si dovrà specificare obbligatoriamente alcun indirizzo. Questo verrà scelto automaticamente dal sistema (allocazione automatica).

### 3.4.2 Ragguppamento dei Simboli

All'occorrenza, i simboli possono essere raggruppati per facilitare la lettura del programma. Per fare questo è sufficiente premere il pulsante destro del mouse in modo da aggiungere un nuovo gruppo all'Editore dei Simboli, quindi con una operazione di "Trascina e Rilascia", spostare i simboli desiderati nella cartella del nuovo gruppo:



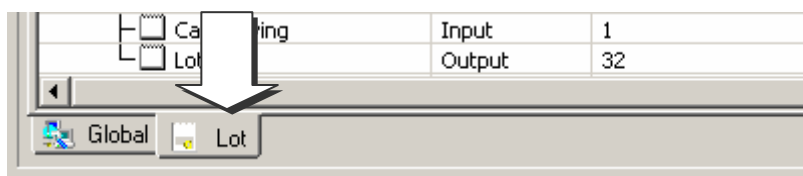
All'interno del programma, il nome del gruppo *LotOne* precede il nome del simbolo *Lot\_full* ed i due identificatori sono separati da un punto.

### 3.4.3 Ambito dei Simboli

I simboli sono normalmente conosciuti da un solo file (ambito locale). Quando si apre un file di programma con un editore, viene aperto anche l'Editore dei Simboli con l'elenco dei simboli specifici per quel file:

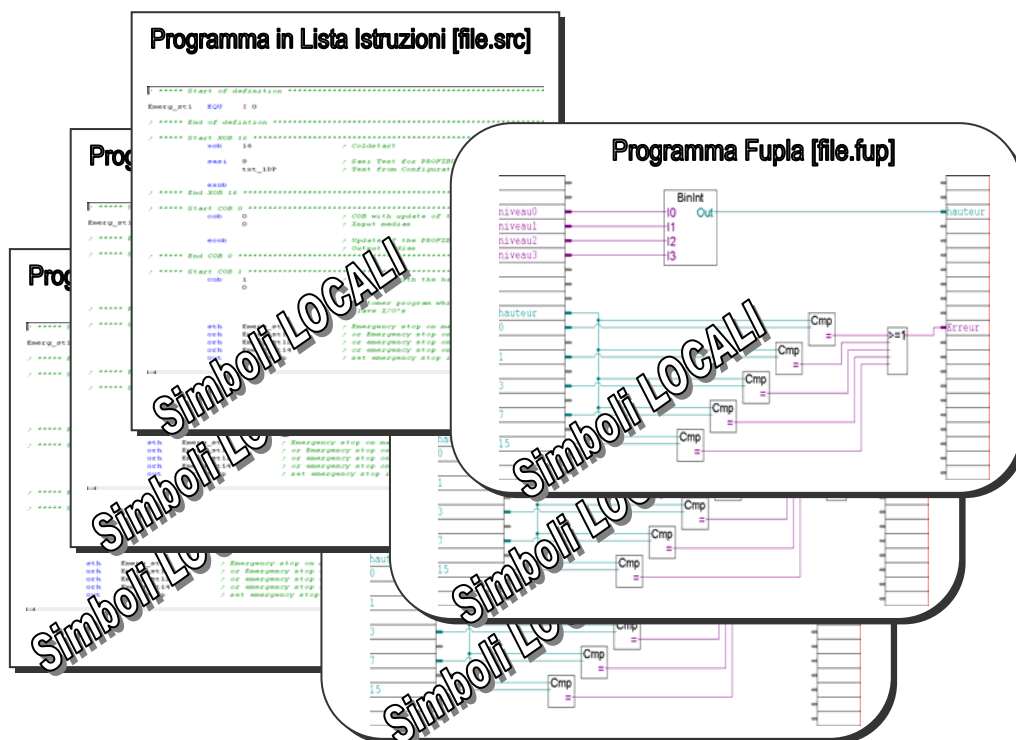
#### Esempio:

Aprendo il file di programma denominato *Lot.src*, si avrà l'apertura automatica dell'Editore dei Simboli avente lo stesso nome.



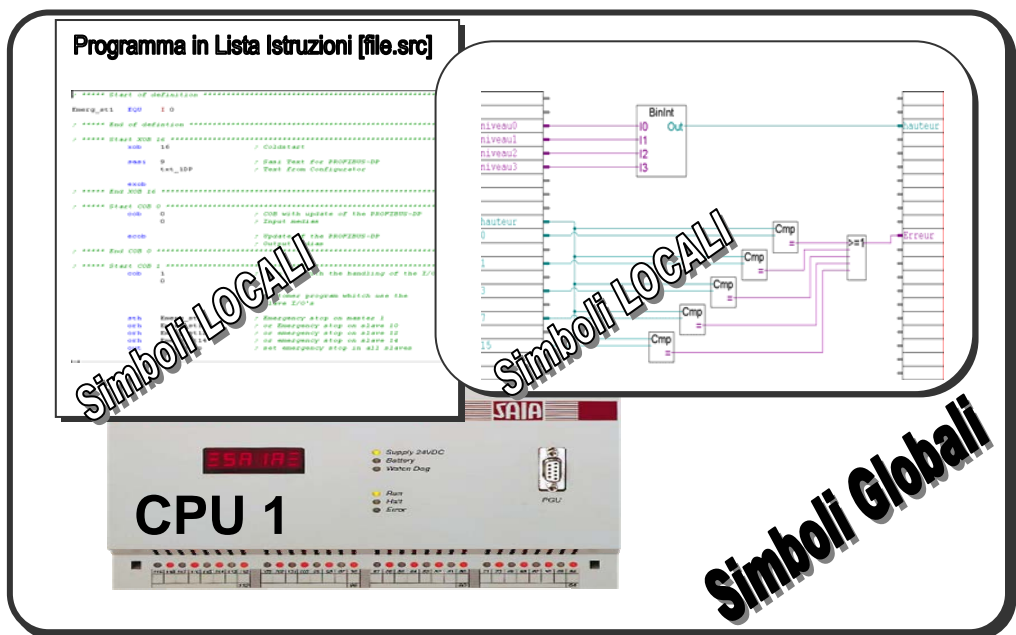
### 3.4.4 Simboli Locali

I Simboli Locali sono conosciuti solo dal file a cui i simboli stessi appartengono.



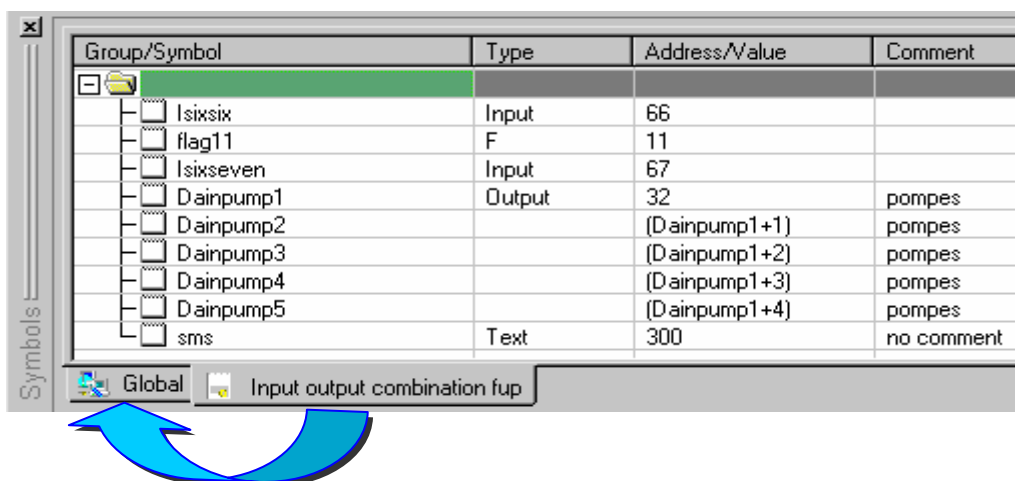
### 3.4.5 Simboli Globali

I simboli globali sono conosciuti da tutti i file della CPU.



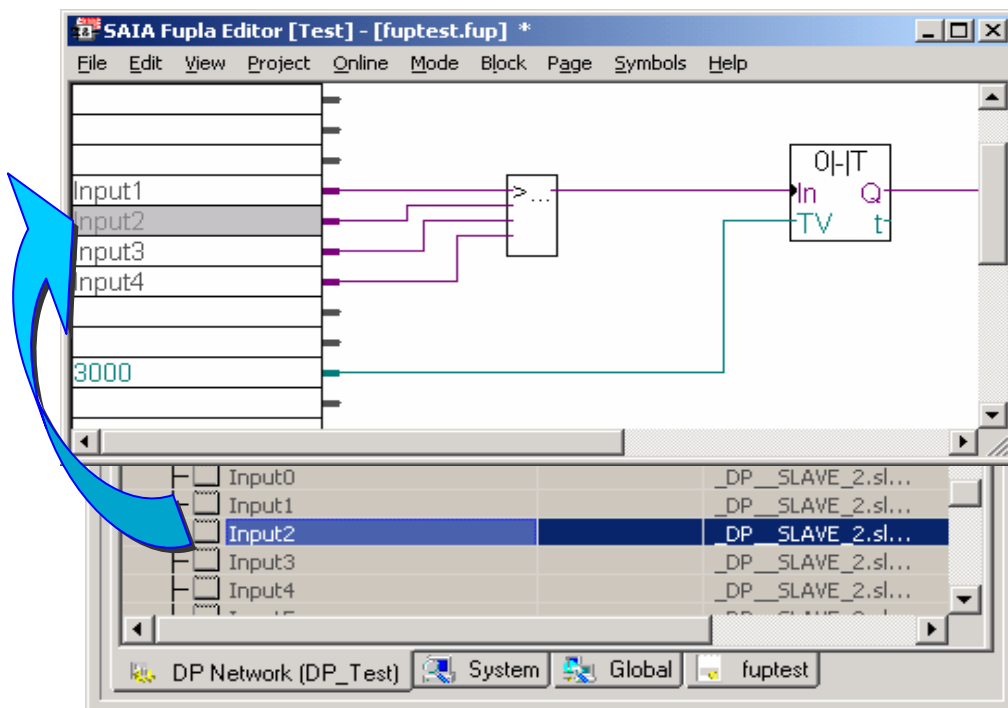


### 3.4.6 Definizione di un Simbolo Globale



Nel caso in cui si desidera utilizzare gli stessi simboli in più file, sarà necessario spostare i simboli dall'elenco locale all'elenco *Globale* (per fare questo sarà sufficiente contrassegnare i simboli desiderati nell'elenco ed utilizzare la funzione *Advanced, Make Global (Avanzato, Rendi Globale)* (pulsante destro del mouse)). Quando un simbolo viene indicato come *Globale*, sarà possibile accedere a tale simbolo da un qualsiasi file appartenente alla CPU del progetto.

### 3.4.7 Definizione dei simboli utilizzati per le reti di comunicazione

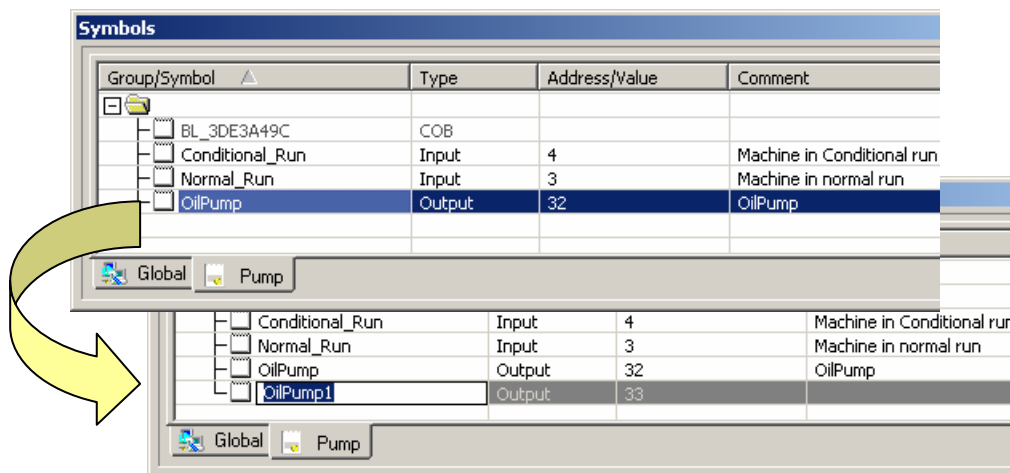


La condivisione di dati tra due differenti PCD è invece un'operazione più complicata rispetto alla condivisione di informazioni tra file. Sarà infatti necessario collegare in rete i due PCD. Tale connessione in rete potrà essere realizzata con i nostri configuratori di rete (attualmente le reti supportate sono: Sbus, Profibus DP, Profibus FMS e LON). Il configuratore di rete elenca tutti i simboli nella lista "Network", e tali simboli potranno essere utilizzati nel programma per spostare dei dati da un PCD ad un altro.

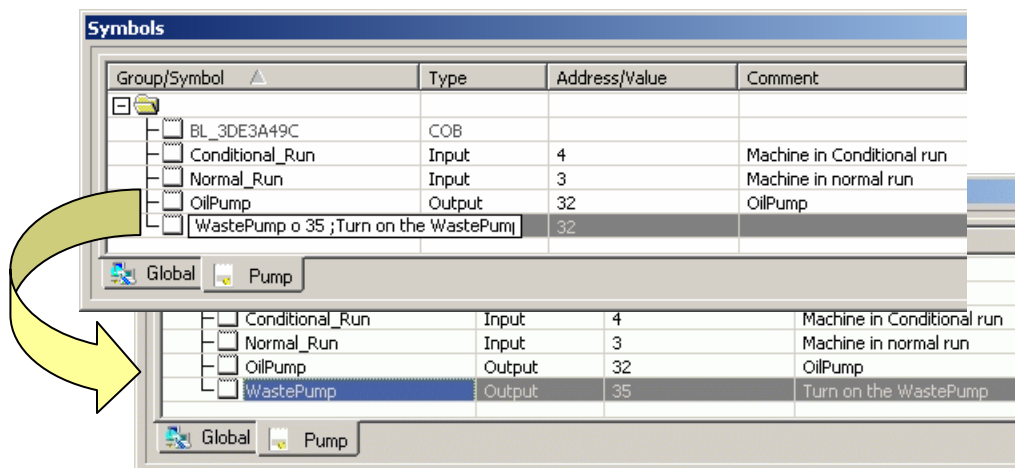
## 3.5 Come lavorare con i simboli

### 3.5.1 Scrittura di un Elenco di Simboli

Aprire il file con il quale si desidera lavorare. Questo determinerà anche l'apertura dell'Editore dei Simboli. Fare clic su Group/Symbol, e premere il tasto *Insert* (*Inserimento*). Verrà aggiunto immediatamente il nuovo campo simbolo all'elenco. Inserire i campi: nome simbolo, tipo, indirizzo/ valore e commento. Premere *Enter* per confermare i dati inseriti.



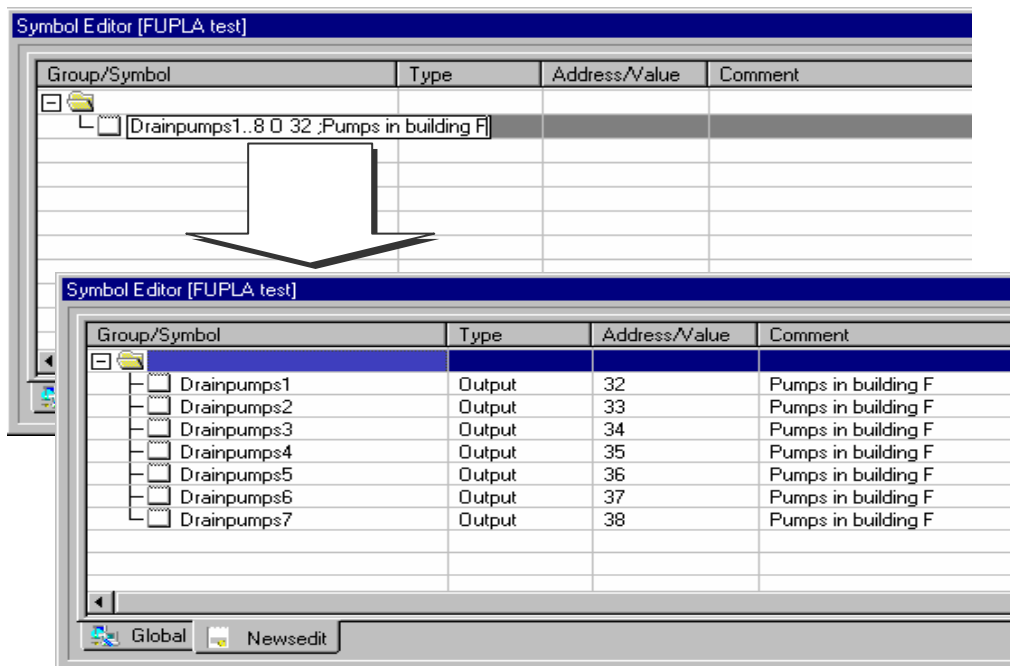
Quindi premere nuovamente il tasto *Insert* (*Inserimento*) per inserire il simbolo successivo. L'editore propone automaticamente il nome simbolo e l'indirizzo precedenti, ma incrementati di uno (vedere la figura in basso). È possibile cambiare il commento ed accettare il nome, il tipo e l'indirizzo/valore, oppure riscrivere tutti i campi con un nuovo nome, tipo, indirizzo/valore e commento.



Se è già stato inserito un elenco, e si vuole semplicemente aggiungere un nuovo simbolo, è sufficiente fare clic su Group/Symbol, e premere il tasto *Insert* (*Inserimento*). In questo modo verrà aperto un nuovo campo simbolo.

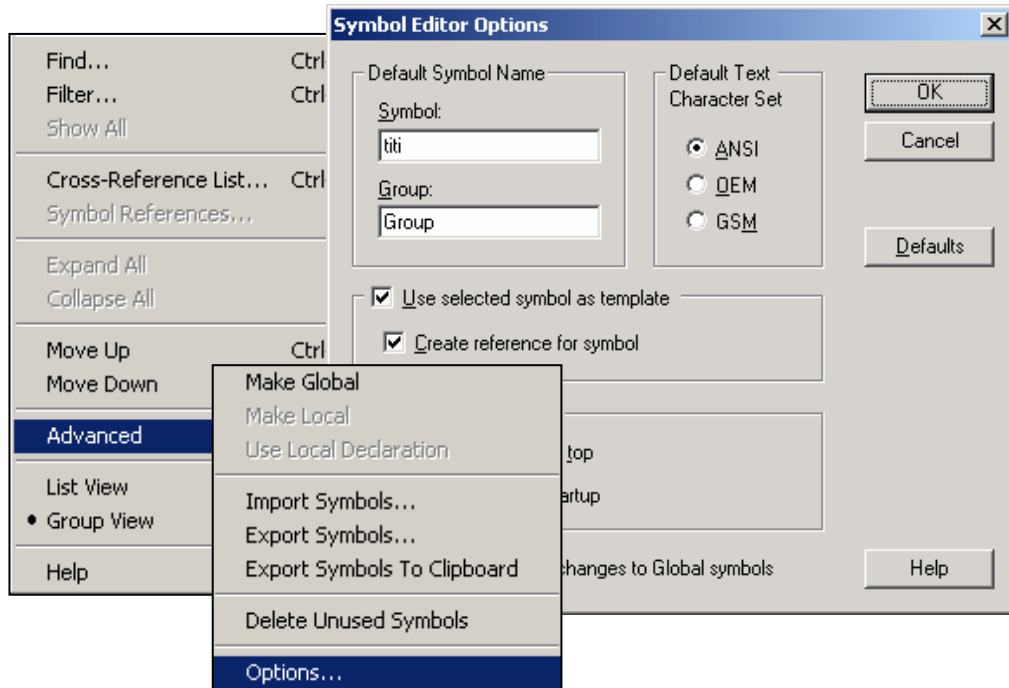
### 3.5.2 Aggiunta di più simboli all'Editore dei Simboli

All'occorrenza, è possibile aggiungere un insieme di simboli ad un elenco esistente. Per fare questo è sufficiente inserire il nome del simbolo con il primo e l'ultimo indirizzo di elemento, come illustrato in figura (Drainpumps1..8 O 32; Pumps in building F). 8 è il numero dei simboli, O sta per uscita, 32 è l'indirizzo di partenza dell'insieme di simboli che si vuole inserire. Premendo il pulsante *Enter*, l'Editore dei Simboli provvederà a completare l'elenco.



### 3.5.3 Simboli di Riferimento

Un elenco di simboli può anche fare riferimento ad un simbolo particolare. Selezionare il menu *Symbols, Advanced, Options...* (*Simboli, Avanzato, Opzioni...*) per aprire la finestra *Symbol Editor Options* (*Opzioni dell'Editore Simboli*), inserire il simbolo e selezionare la casella *Create reference for symbol* (*Crea un riferimento per il simbolo*). Confermare con *OK*.

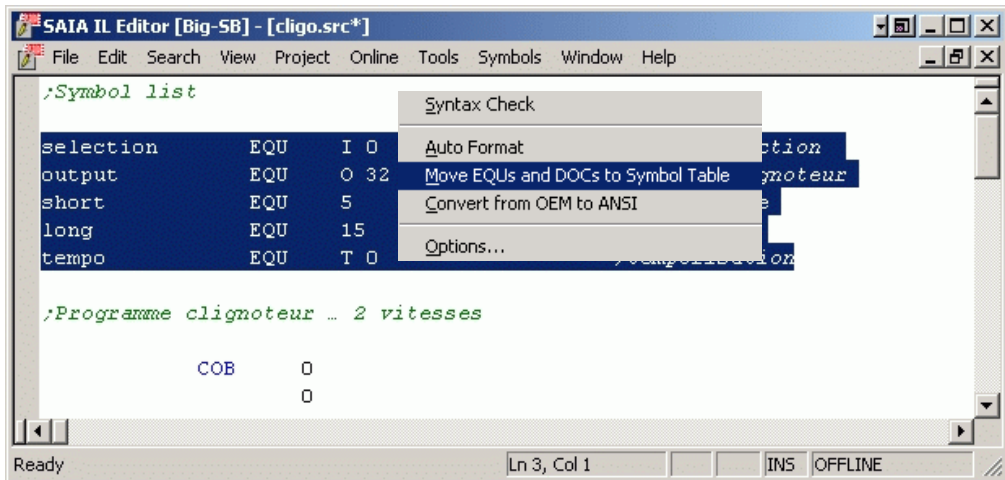


Un'altra opzione disponibile è quella di inserire un simbolo ed un indirizzo nell'Editore dei Simboli. Aprire la finestra Opzioni, (selezionando la voce di menu *Symbols, Advanced* (*Simboli, Avanzati*) e quindi *Options* (*Opzioni*)) inserire il simbolo e selezionare *Create reference for symbol* (*Crea un riferimento per il simbolo*). Fare clic su *OK* ed evidenziare il simbolo nell'Editore dei Simboli. Premere il tasto *Insert* (*Inserimento*); il simbolo viene inserito ma incrementato di 1. Questo può essere utile nel caso in cui si abbia una stringa di ingressi e di uscite e si desideri cambiare il loro indirizzo fisico a livello software; in questo modo sarà sufficiente cambiare il primo indirizzo, e tutti gli altri verranno cambiati di conseguenza.

Group/Symbol	Type	Address/Value	Comment
DrainPumps1	Output	32	Pumps in building F
DrainPumps2	Output	(DrainPumps1+1)	Pumps in building F
DrainPumps3	Output	(DrainPumps1+2)	Pumps in building F
DrainPumps4	Output	(DrainPumps1+3)	Pumps in building F
DrainPumps5	Output	(DrainPumps1+4)	Pumps in building F
DrainPumps6	Output	(DrainPumps1+5)	Pumps in building F
DrainPumps7	Output	(DrainPumps1+6)	Pumps in building F
DrainPumps8	Output	(DrainPumps1+7)	Pumps in building F

### 3.5.4 Come importare dei simboli da istruzioni "EQUATE"

Se si dispone di vecchi file in Lista Istruzioni creati con PG4/3 e contenenti istruzioni EQU o DOC, per importare i corrispondenti simboli è sufficiente selezionare le suddette istruzioni quindi usare la voce di menu: *Tools, Move EQUs and Docs to Symbol Table (Strumenti, Sposta EQU e Doc nella Tabella Simboli)*. I simboli verranno spostati dal file di programma all'elenco dei simboli.

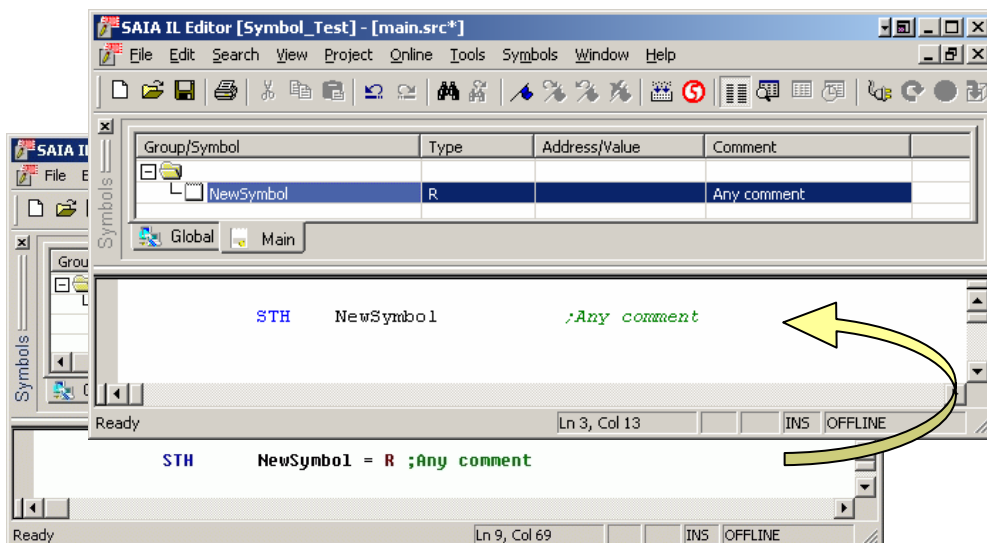


### 3.5.5 Come importare dei simboli da un'altra applicazione

È possibile anche importare dei simboli da un altro programma (CAD Elettrico) ed utilizzarli all'interno del progetto. Questo rende congruente la documentazione dell'intero progetto e farà sì che vengano utilizzate le stesse etichette negli schemi elettrici ed all'interno del programma. Per fare questo è sufficiente utilizzare la funzione di Esportazione del programma CAD per esportare i simboli in un file di testo, quindi importarli nell'Editore dei Simboli.

### 3.5.6 Aggiunta di simboli durante la scrittura del programma in Lista Istruzioni

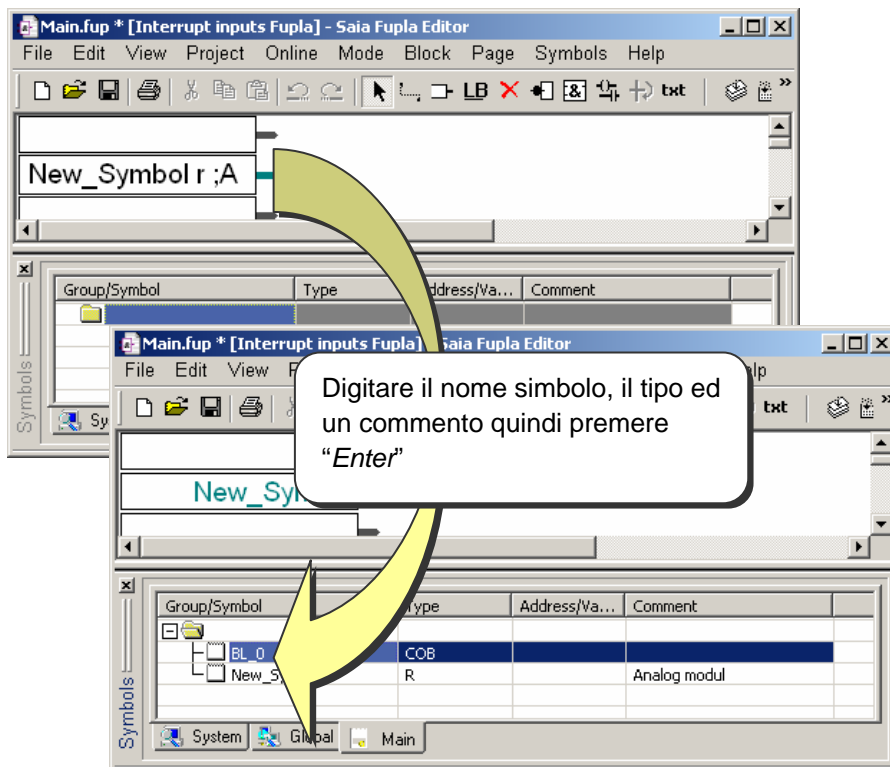
Durante la scrittura del programma, ogni volta che si desidera aggiungere un nuovo simbolo è sufficiente aggiungere alla riga la stringa "*= type address ;comment*" (*= tipo indirizzo ;commento*). Alla pressione del tasto *Enter* la definizione del simbolo verrà spostata all'interno dell'elenco dei simboli. Ad esempio:



### 3.5.7 Aggiunta di simboli durante la scrittura del programma in FUPLA

L'editore FUPLA funziona esattamente allo stesso modo. Si possono quindi inserire nuovi simboli nell'elenco dei simboli direttamente dai campi di Input/Output FUPLA.

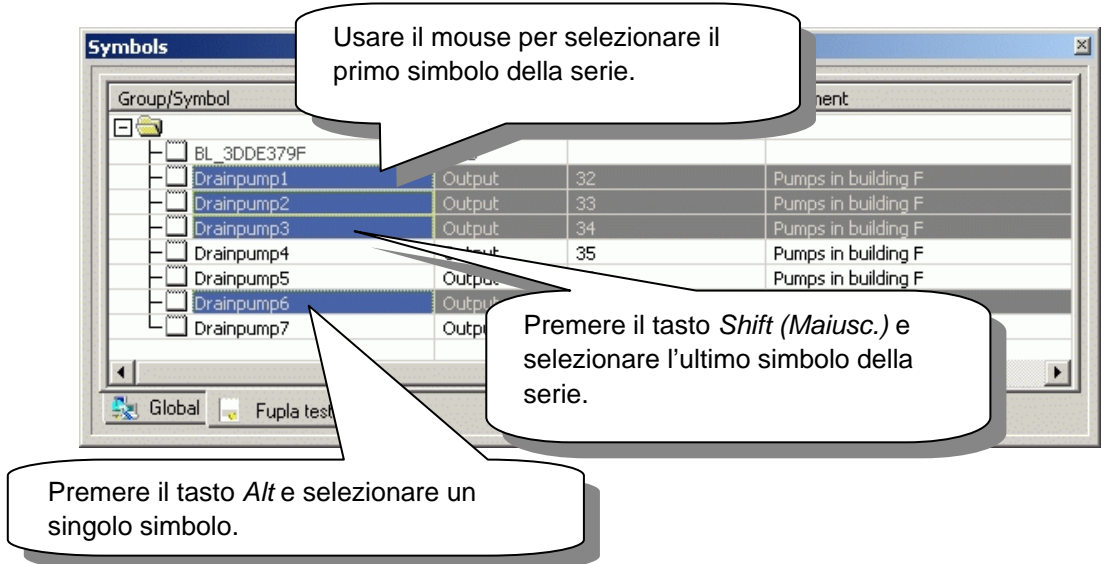
Sintassi: *Tipo simbolo [Indirizzo] [;Commento]*



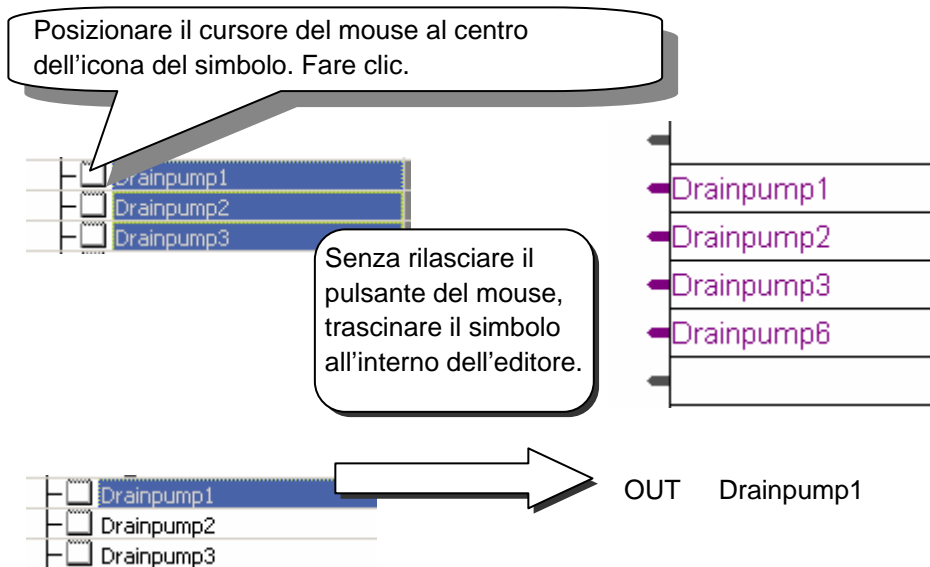
### 3.5.8 Trasferimento di simboli

Per evitare di inserire più volte all'interno di un programma i nomi dei simboli (correndo il rischio di commettere errori di digitazione) è possibile selezionare uno o più simboli all'interno dell'Editore dei Simboli e trascinarli all'interno del programma Fupla o IL.

L'esempio seguente illustra la selezione di più simboli



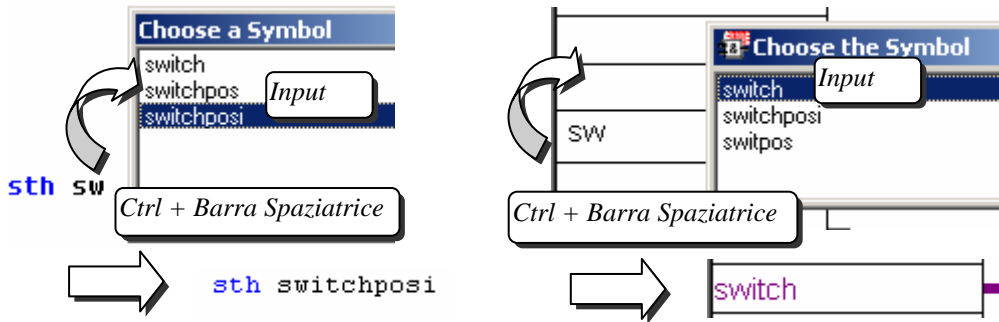
L'esempio seguente illustra come trascinare i simboli all'interno dell'editore Fupla o IL



### 3.5.9 Completamento Automatico dei Simboli

Se per i simboli si utilizzano dei nomi lunghi, la lettura dei programmi risulterà facilitata. Per contro sarebbe alquanto frustrante dover reinserire ripetutamente questi nomi lunghi nel programma. Questa operazione può essere evitata inserendo semplicemente le prime lettere di un simbolo, e ricercare tutti i simboli che contengono queste lettere utilizzando i tasti "Ctrl + Barra Spaziatrice".

**Esempio:**



### 3.5.10 Allocazione Automatica

Finora abbiamo sempre dichiarato gli elementi nel modo seguente:

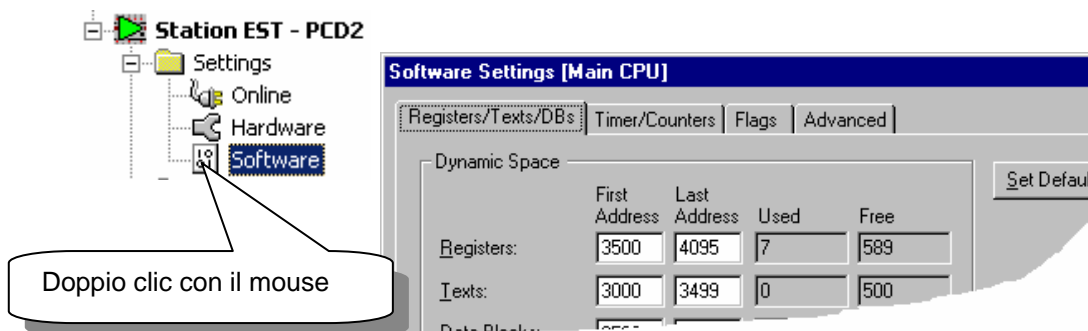
Nome Simbolo                      Tipo                      Indirizzo                      Commento

**Esempio:**    VelocitàPompa                      R                      2000                      ; Velocità in l/min

Quando si inserisce un qualsiasi tipo di simbolo, che non sia un ingresso o una uscita, non sarà necessario inserire l'indirizzo. In questo caso PG5 provvederà ad assegnare un indirizzo all'elemento durante la costruzione del programma (Build). Questa funzione è definita autoallocazione. PG5 ricercherà nei parametri software il campo di indirizzi configurato per quel tipo di elemento, e provvederà ad assegnare uno di questi indirizzi durante il processo di costruzione del programma.

**Esempio:**    VelocitàPompa                      R                      ; Velocità in l/min

Nel programma viene dichiarato un registro privo di indirizzo:



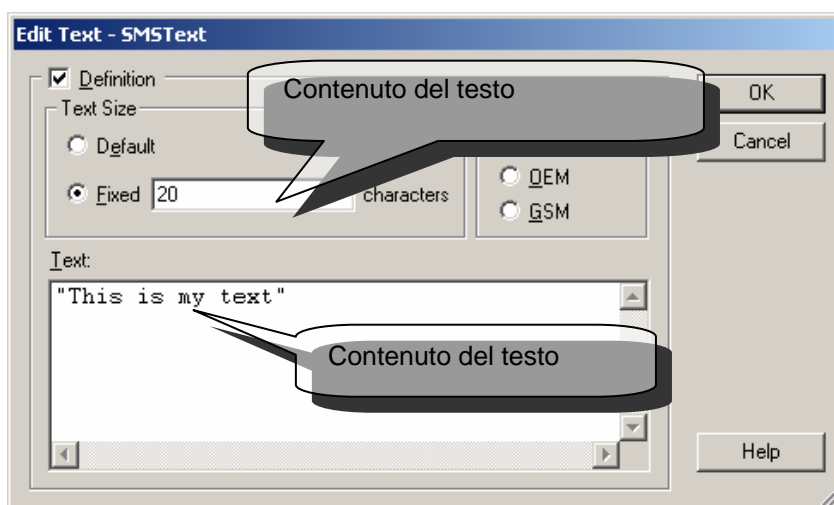
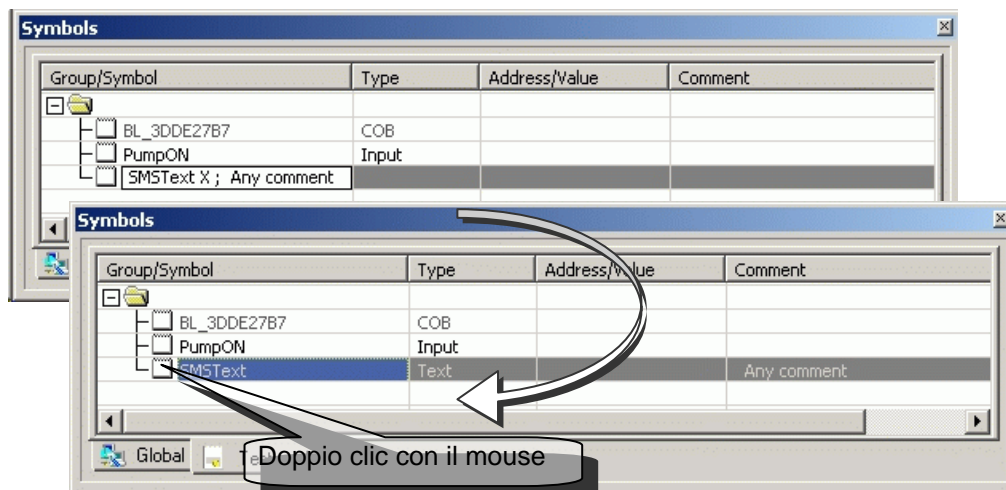
Al registro verrà assegnato un numero compreso tra 3500 e 4095 durante il processo di costruzione, in quanto nelle impostazioni *Software (Software settings)* lo spazio dinamico dichiarato per i registri è quello compreso tra 3500 e 4095.



### 3.5.11 Inserimento di testi

Per aggiungere un testo nel PCD è necessario aver dichiarato tale testo in precedenza. Per fare questo è sufficiente inserire una X dopo il nome del simbolo:

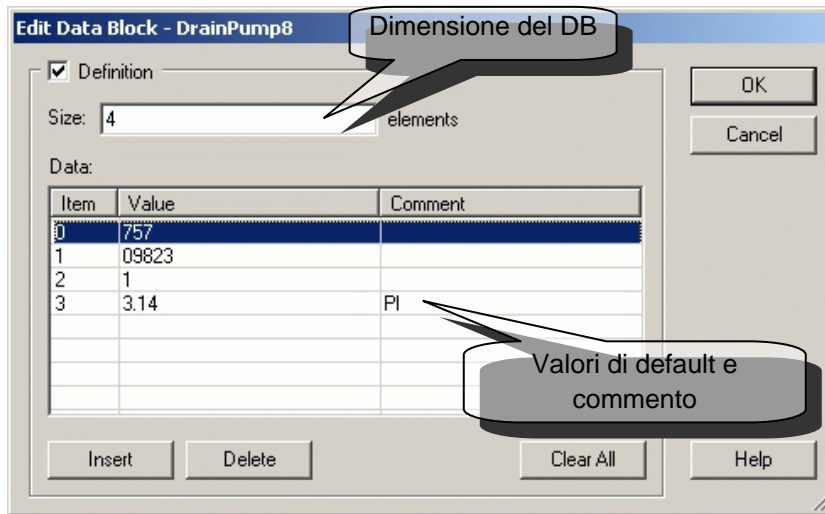
**Esempio:**



Non dimenticare di racchiudere il testo tra " ", altrimenti il testo non risulterà valido.

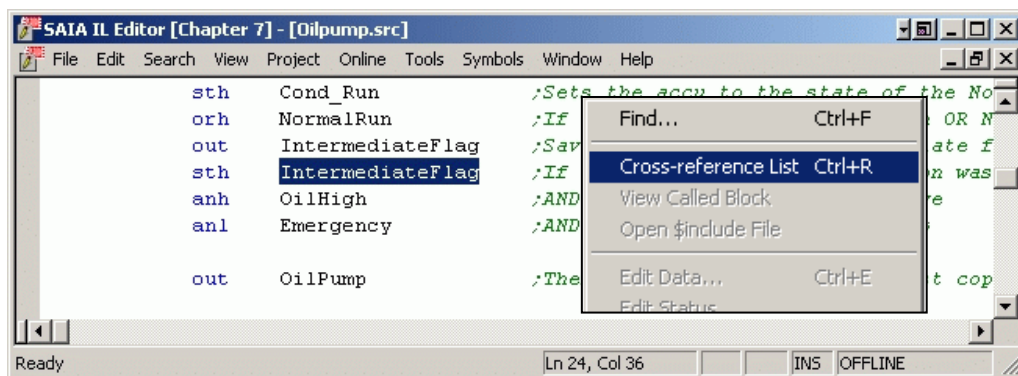
### 3.5.12 Inserimento di DB

Anche per i DB è previsto un editore speciale. Consultare l'help per maggiori informazioni

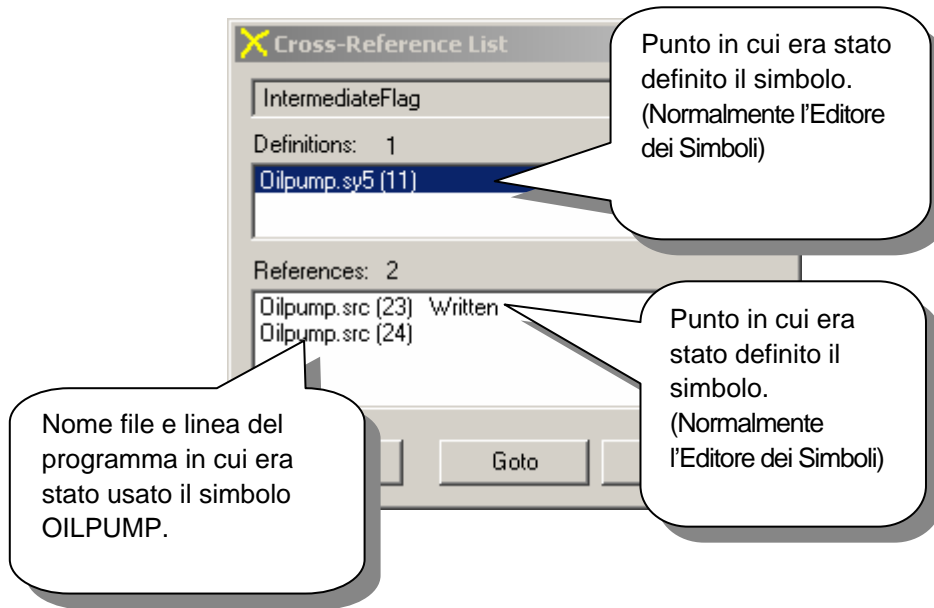


### 3.5.13 Ricerca di un Simbolo

Spesso un simbolo viene utilizzato diverse volte all'interno di un file di programma o anche all'interno di file differenti. Dopo aver eseguito con successo la costruzione del programma (Build) è possibile fare clic con il pulsante destro del mouse su un qualsiasi simbolo per attivare la funzione *Cross reference List* (Lista riferimenti Incrociati).

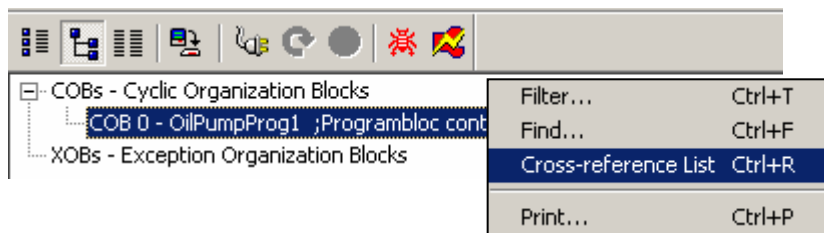


La funzione "Riferimenti Incrociati" fornisce un elenco contenente: nome file e numero di riga oltre al numero di volte in cui un determinato simbolo è stato utilizzato. Facendo doppio clic su uno qualsiasi degli elementi dell'elenco verrà aperto il file del programma con il cursore posizionato sul simbolo in questione.



Lo strumento "riferimenti incrociati" è disponibile non solo in S-Edit e Fupla ma anche in varie altre visualizzazioni del Project Manager.

**Esempio:** Vista struttura blocchi

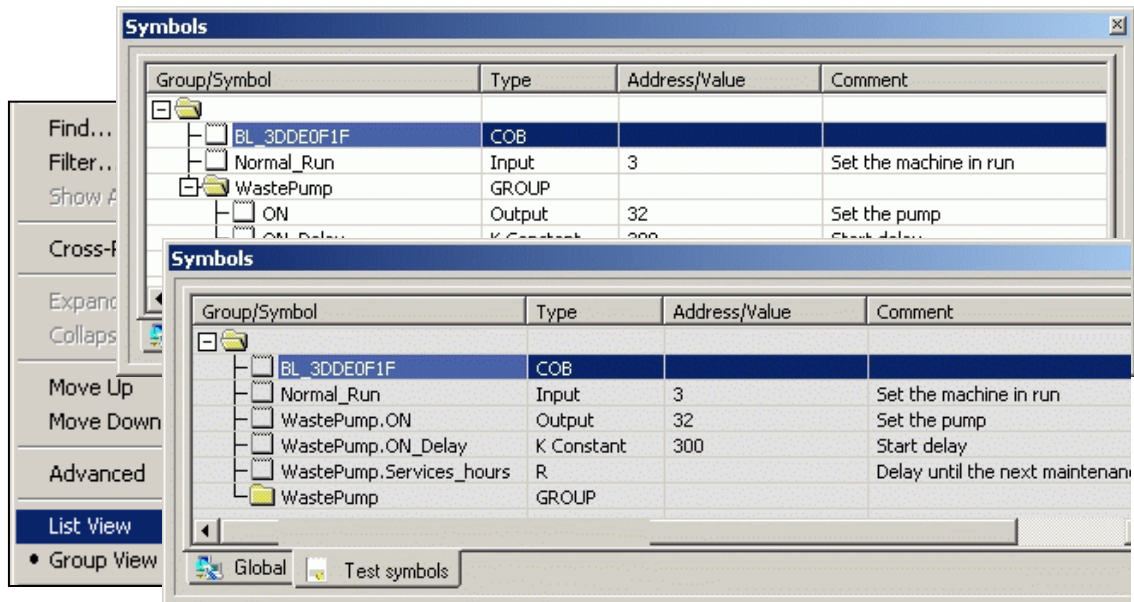


### 3.5.14 Disposizione dei Simboli

I simboli vengono elencati nell'ordine in cui sono stati inseriti. In questo modo, i simboli che erano stati inseriti nello stesso tempo resteranno raggruppati anche se successivamente sono stati aggiunti nuovi simboli.

### 3.5.15 Riorganizzazione in “List View” (Disposizione a Lista)

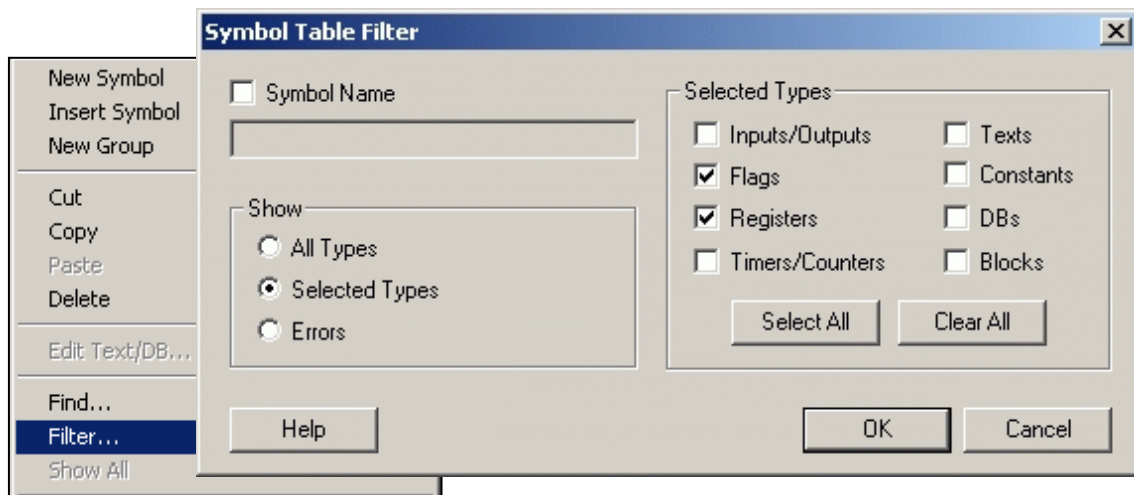
I simboli possono essere riorganizzati passando semplicemente dalla disposizione a gruppo (*Group view*) alla disposizione a lista (*List view*). È sufficiente fare clic su una delle voci di intestazione per poter organizzare i simboli in base ai campi Nome, Tipo, Indirizzo o Commento.



#### Visualizzazione con Filtro

Ritornando alla disposizione Group View viene ristabilito il vecchio ordine.

Quando l'elenco comprende molti simboli, può essere utile visualizzare solo certi simboli, oppure solo i simboli con un certo nome.



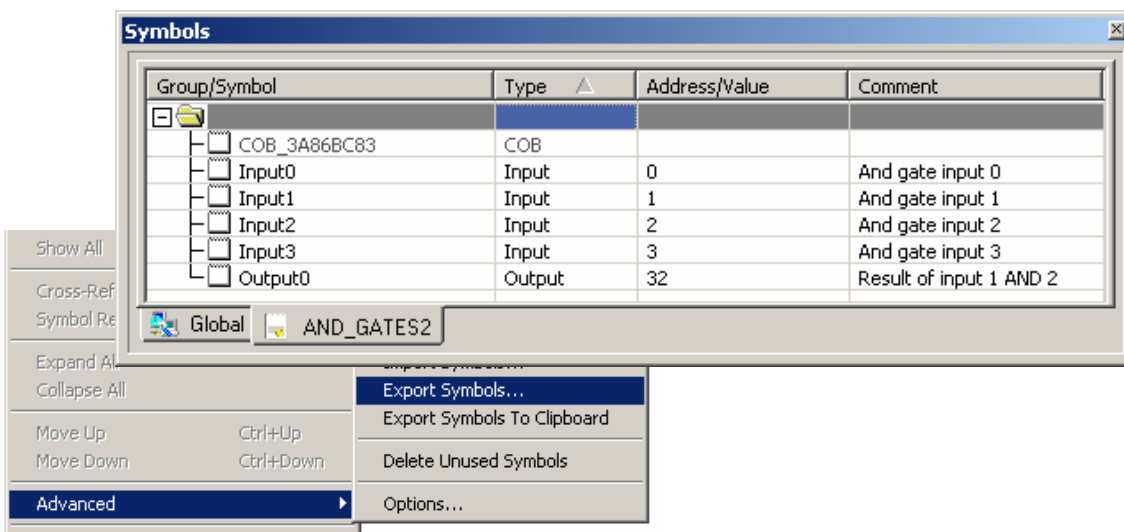
La funzione filtro permette di selezionare la vista desiderata. Quando il filtro è attivo l'icona del simbolo assume un aspetto diverso.

**Esempio :** L'Editore dei Simboli visualizza tutti gli Elementi  
 Filtro attivo > Non tutti i simboli sono visualizzati =>

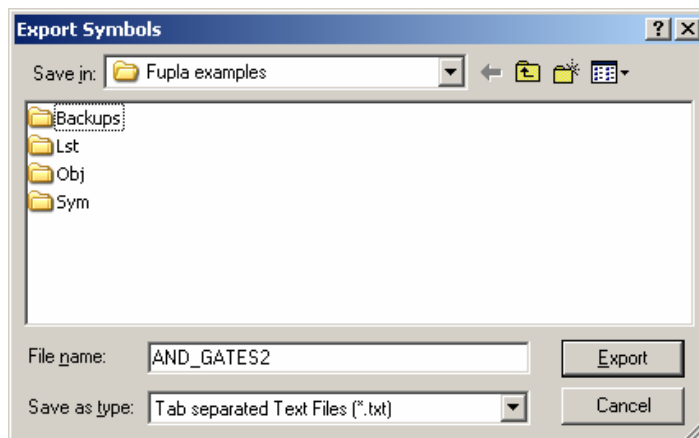
### 3.5.16 Esportazione di simboli

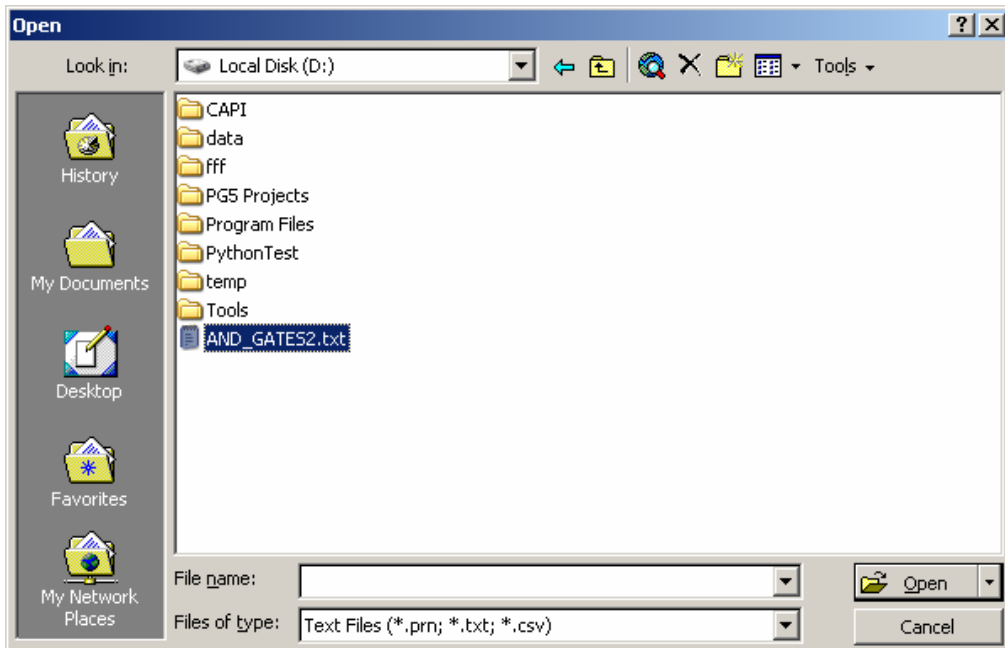
L'elenco dei simboli di un programma può essere esportato per essere usato all'interno di altre applicazioni (quali Excel, Visiplus o Word) ad esempio per generare un rapporto di messa in servizio.

L'esempio seguente illustra l'esportazione dei simboli per l'uso nell'applicativo Excel: All'interno dell'editore dei simboli, selezionare la voce *Export Symbols (Esporta Simboli)* dal menu contestuale.



Quando si esporta un elenco simboli per l'uso in Excel, si raccomanda di selezionare il formato *Tab separated Text file (\*.txt)*. Si otterranno risultati migliori anche rispetto all'esportazione in formato file *Excel (\*.xls)*.





Avviare Excel ed aprire il file di testo contenente i simboli esportati.

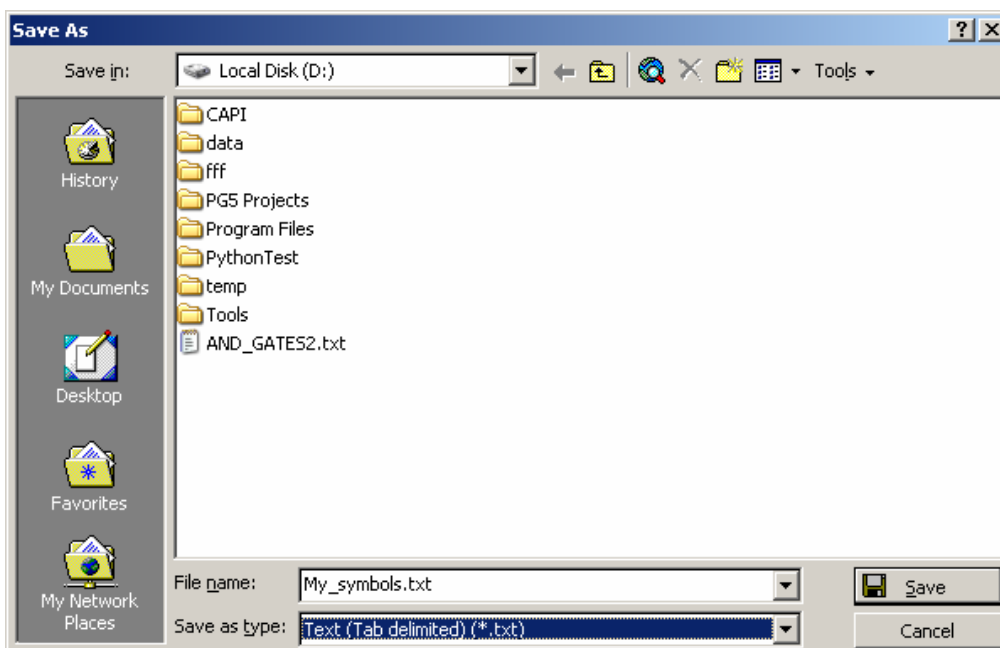
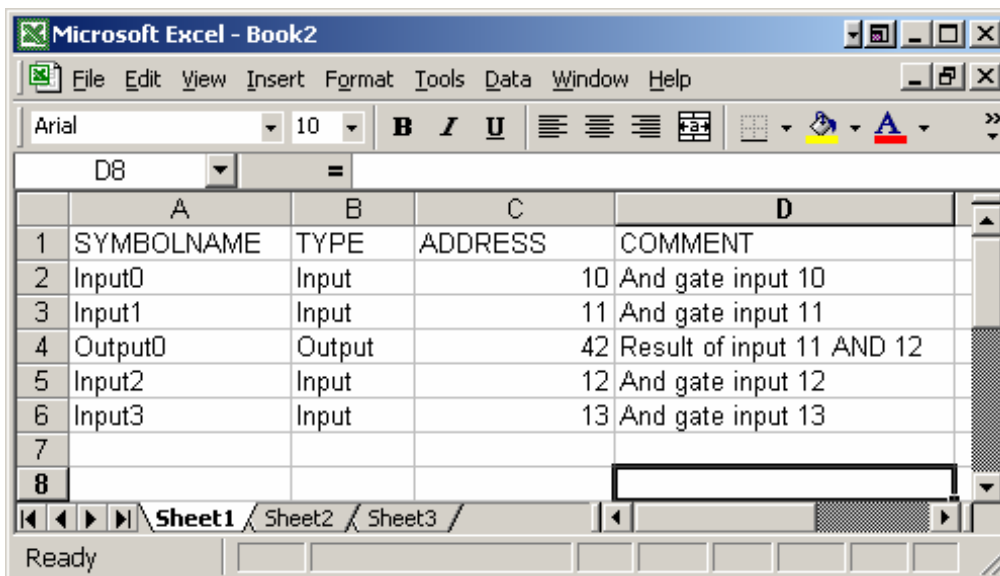
The screenshot shows Microsoft Excel with the file 'AND\_GATES2.txt' open. The spreadsheet contains the following data:

	A	B	C	D
1	SYMBOLNAME	TYPE	ADDRESS	COMMENT
2	Input0	Input		0 And gate input 0
3	Input1	Input		1 And gate input 1
4	Output0	Output		32 Result of input 1 AND 2
5	Input2	Input		2 And gate input 2
6	Input3	Input		3 And gate input 3
7	COB_3A86BC83	COB		
8				

The status bar at the bottom shows 'Ready' and the active sheet is 'AND\_GATES2'.

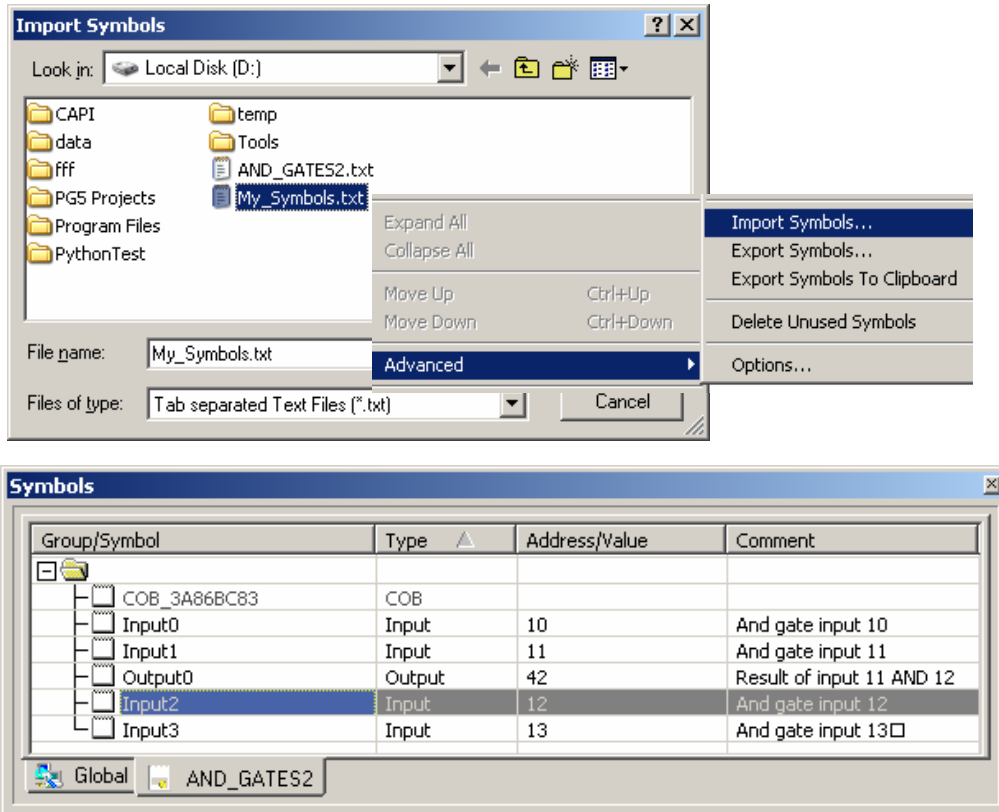
### 3.5.17 Importazione di simboli

E' anche possibile scrivere un elenco di simboli utilizzando l'applicativo Excel e quindi importarlo all'interno di un progetto PG5. Per eseguire tale operazione, generare un file di simboli come sotto illustrato quindi salvarlo in formato *Text (Tab delimited) (\*.txt)*.



Nell'editore di simboli PG5, selezionare la voce *Advanced, Import Symbols* (*Avanzate, Importa Simboli*) all'interno del menu contestuale quindi selezionare il file interessato ed importarlo.

**In caso si verifichi un qualsiasi problema, accertarsi che il file Excel sia stato opportunamente chiuso.**





### 3.5.18 Inizializzazione dei simboli

Esistono due modi per inizializzare i simboli utilizzati dal PCD:

- inizializzazione durante una partenza a freddo del PLC (all'accensione)
- inizializzazione durante il trasferimento del programma al PCD (download)

#### Durante la partenza a freddo

L'inizializzazione dei simboli durante una partenza a freddo viene eseguita all'interno del blocco XOB 16. Questo blocco viene elaborato una sola volta, durante la partenza a freddo del PCD. L'utente scrive pertanto il codice IL relativo all'inizializzazione dei simboli all'interno del blocco XOB 16.

**Esempio:** inizializzazione di una flag e di un registro durante la partenza a freddo del PCD

#### Programma in IL

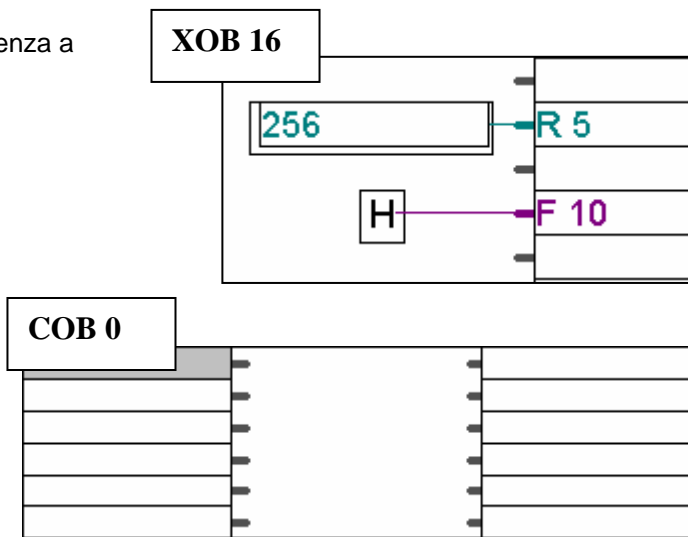
```
XOB 16 ; Blocco di partenza a
        freddo
LD R 5 ; R 5 = 256
    256
SET F 10 ;F 10 = 1

EXOB

COB 0 ; Blocco ciclico
    0
...
; Il vostro programma
...

ECOB
```

#### Programma in Fupla



Per maggiori dettagli sui blocchi COB e XOB, consultare il capitolo 5 del presente documento.

#### Durante il trasferimento del programma al PCD (Download)

Per inizializzare un simbolo durante il trasferimento (download) del programma al PCD, l'indirizzo del simbolo deve essere seguito dai caratteri := (due punti, uguale), seguiti a loro volta dal valore cui il simbolo deve essere inizializzato.

#### Esempio:

Group/Symbol	Type	Address/Value	Com
BL_3E315C55	COB	0	
SymbolA	R	5:= 256	
SymbolB	F	10:=1	



#### Attenzione

Ricordarsi di selezionare la seguente casella prima di procedere al trasferimento del programma:



First-time Initialisation Data

### 3.5.19 Nomi dei Simboli

I nomi dei Simboli sono i nomi che possono essere assegnati agli elementi del PCD (ingressi, uscite, flag, registri, COB ecc). I nomi dei simboli possono essere lunghi fino ad 80 caratteri e possono essere indifferentemente scritti in maiuscolo o minuscolo, a meno che non contengano caratteri speciali. MotorOn equivale a MOTORON, ma GRÜN non equivale a grün.

I simboli devono iniziare con una lettera ( a-z, A-Z); non possono iniziare con un numero. All'interno del simbolo si possono mischiare a piacimento numeri, lettere e segni di sottolineatura "\_". Il nome del simbolo non può tuttavia contenere spazi.

**Non è possibile utilizzare parole riservate come nomi di simboli.**

### 3.5.20 Parole Riservate

Le seguenti parole sono riservate e non possono essere utilizzate come nomi di simboli:

- Dichiarazioni assembler, quali PUBL, EXTN, EQU, DEF, LEQU, LDEF; MACRO, ENDM, EXITM ecc.
- Codici di controllo supporto e tipi di dati (I, O, F, R, C, T, K, M, COB, FB, TEXT, X, SEMA, DB).
- Codici speciali dell'istruzione MOV (N, Q, B, W, L, D).
- Codici di condizione (H, L, P, N, Z, E).
- Tutti i codici mnemonici delle istruzioni.
- Istruzioni CASI, CRD, CRW (non implementate)
- Simboli predefiniti.
- Simboli interni usati per l'allocazione automatica delle risorse, che iniziano con un carattere di sottolineatura, ad esempio:  
\_\_\_\_TEXT, \_\_\_\_\_F.
- Simbolo interno \_\_CSTART\_\_, usato per le assegnazioni \$\$.

## Indice

---

4	Programmazione FUPLA	4-3
4.1	Introduzione	4-3
4.2	Preparazione di un progetto Fupla	4-4
4.2.1	Creare il nuovo progetto	4-4
4.3	Organizzazione di una finestra Fupla	4-5
4.4	Scrittura dei Simboli	4-6
4.4.1	Come aggiungere un nuovo Simbolo nella finestra Simboli	4-7
4.4.2	Modalità di indirizzamento dei simboli	4-8
4.4.3	Utilizzo di un Simbolo dell'elenco Simboli in un programma FUPLA	4-9
4.4.4	Simboli locali e globali	4-10
4.5	Editare i connettori	4-11
4.5.1	Inserire i connettori	4-11
4.5.2	Modificare il simbolo interno ad un connettore	4-11
4.5.3	Inserire in modo rapido un simbolo e il connettore associato	4-11
4.5.4	Trascinare, Copiare/Incollare e Cancellare un simbolo	4-12
4.5.5	Copiare/Incollare e Cancellare un connettore	4-12
4.5.6	Estendere i connettori	4-12
4.5.7	Spostare verticalmente un connettore	4-12
4.6	Scrittura di una funzione Fupla	4-13
4.6.1	Selettore FBox	4-13
4.6.2	Scrittura FBox	4-14
4.6.3	Scrittura di Fbox estendibili	4-14
4.6.4	Inversione logica	4-14
4.6.5	Dinamizzazione	4-15
4.6.6	Commenti	4-15
4.6.7	Aiuto sugli FBox	4-15
4.7	Collegare FBox e connettori	4-16
4.7.1	Collegamento mediante trascinamento dell'FBox	4-16
4.7.2	Collegamento con instradamento automatico	4-16
4.7.3	Collegamento multiplo con instradamento automatico	4-16
4.7.4	Collegare tutti gli Ingressi/Uscite di un FBox a connettori	4-16
4.7.5	Cancellare linee, FBox, connettori o simboli	4-17
4.7.6	Spostare verticalmente FBox/connettori senza rimuovere i collegamenti	4-17
4.7.7	Inserire FBox senza rimuovere i collegamenti	4-17
4.7.8	Regole da osservare	4-17
4.8	Scrittura di pagine Fupla	4-18
4.8.1	Inserimento di pagine	4-18
4.8.2	Cancellazione di una pagina	4-18
4.8.3	Navigazione tra le pagine	4-18
4.8.4	Documentazione della pagina	4-19
4.8.5	Elaborazione del programma da parte del PCD	4-19
4.9	Copia e incolla	4-20
4.9.1	Come copiare e incollare una parte di programma	4-20
4.9.2	Come copiare e incollare simboli	4-20
4.10	Esportazione e importazione di pagine	4-21
4.10.1	Esportazione pagina	4-21
4.10.2	Importazione pagina	4-22
4.11	Scrittura del primo programma Fupla	4-24
4.11.1	Obiettivo	4-24
4.11.2	Metodo	4-24

4.11.3	Programmazione	4-26
4.12	Costruzione (Build) del programma	4-27
4.13	Trasferimento (Download) del programma nel PCD	4-28
4.14	Individuazione e correzione degli errori (Debug)	4-28
4.14.1	Pulsanti On/Offline – Run – Stop - Step-by-step	4-28
4.14.2	Breakpoint (Interruzioni)	4-28
4.14.3	Visualizzazione dei simboli o degli indirizzi	4-29
4.14.4	Visualizzazione dello stato dei simboli in Fupla	4-30
4.14.5	Modifica di simboli „In Linea“	4-30
4.14.6	Visualizzazione dello stato dei simboli con la “Finestra di Monitoraggio” (Watch Window)	4-31
4.14.7	Impostazione dell’orologio PCD	4-31
4.15	Finestra di Configurazione	4-32
4.15.1	Tipi di Parametri Configurazione	4-33
4.15.2	Inizializzazione degli FBox HEAVAC	4-34
4.15.3	FBox HEAVAC con parametri di configurazione	4-35
4.15.4	Applicazione Mini HEAVAC	4-35
4.15.5	Parametri dopo il trasferimento del programma	4-36
4.15.6	Scrittura dei parametri on-line	4-36
4.15.7	Lettura dei parametri on-line	4-37
4.15.8	Ripristino dei parametri di default	4-37
4.15.9	Definizione di simboli per i parametri di configurazione	4-38
4.15.10	Definizione degli indirizzi dei parametri di configurazione	4-39
4.16	Messa in servizio di un modulo analogico	4-40
4.16.1	Acquisizione di una misura analogica	4-40
4.16.2	Esempio per i moduli di ingressi analogici PCD2.W340	4-41
4.16.3	Esempio per i moduli di uscite analogiche PCD2.W610	4-42

## 4 Programmazione FUPLA

---

### 4.1 Introduzione

L'editore Fupla rappresenta l'introduzione più semplice e più rapida alla programmazione dei controllori PCD. Il termine "Fupla" significa "*F*unction *P*LAn", un ambiente di programmazione grafico in cui l'utente può disegnare i programmi con l'ausilio di centinaia di funzioni. Queste funzioni sono organizzate in librerie orientate alle applicazioni di base, con l'aggiunta di funzioni specializzate per determinati ambienti professionali. Queste ultime comprendono una libreria HEAVAC per le applicazioni di riscaldamento, ventilazione e condizionamento; una libreria modem per lo scambio dati tra PLC attraverso linea telefonica, (analogica, ISDN, GSM, GPRS), i messaggi SMS, Pager e DTMF.

Sono anche disponibili altre librerie per reti di comunicazione LON, EIB o prodotti Belimo.

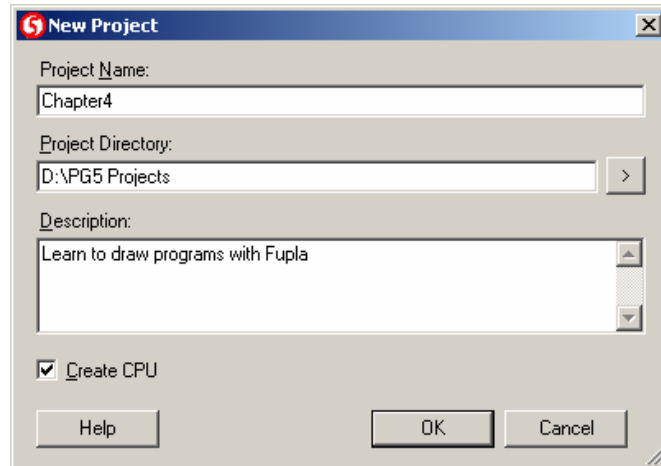
Il principale vantaggio della programmazione Fupla risiede nel fatto che l'utente può mettere in servizio un PCD senza dover scrivere neppure una riga di programma e senza alcuna particolare conoscenza di programmazione.

## 4.2 Preparazione di un progetto Fupla

Per la preparazione di un esempio, si consiglia di creare un nuovo progetto contenente i file necessari per la scrittura del programma Fupla:

### 4.2.1 Creare il nuovo progetto

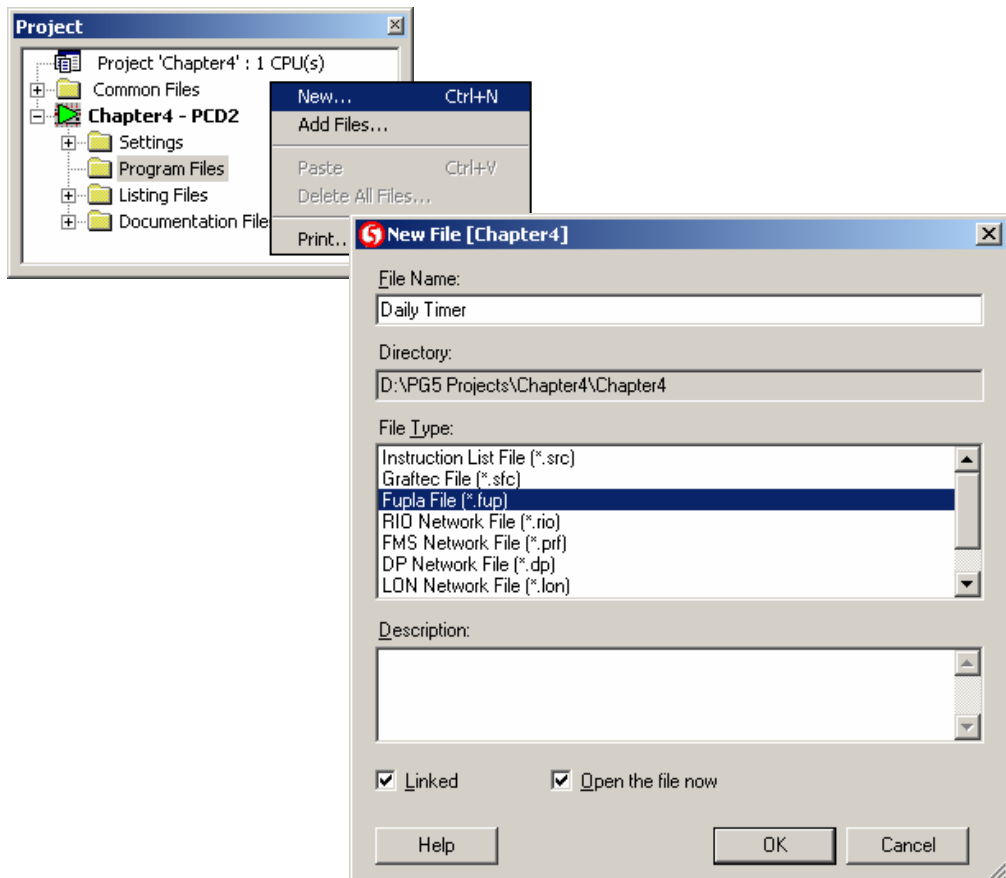
Nella finestra *SAIA Project Manager*, selezionare il comando di menu *Project, New...* (*Progetto, Nuovo*) e creare un nuovo progetto.



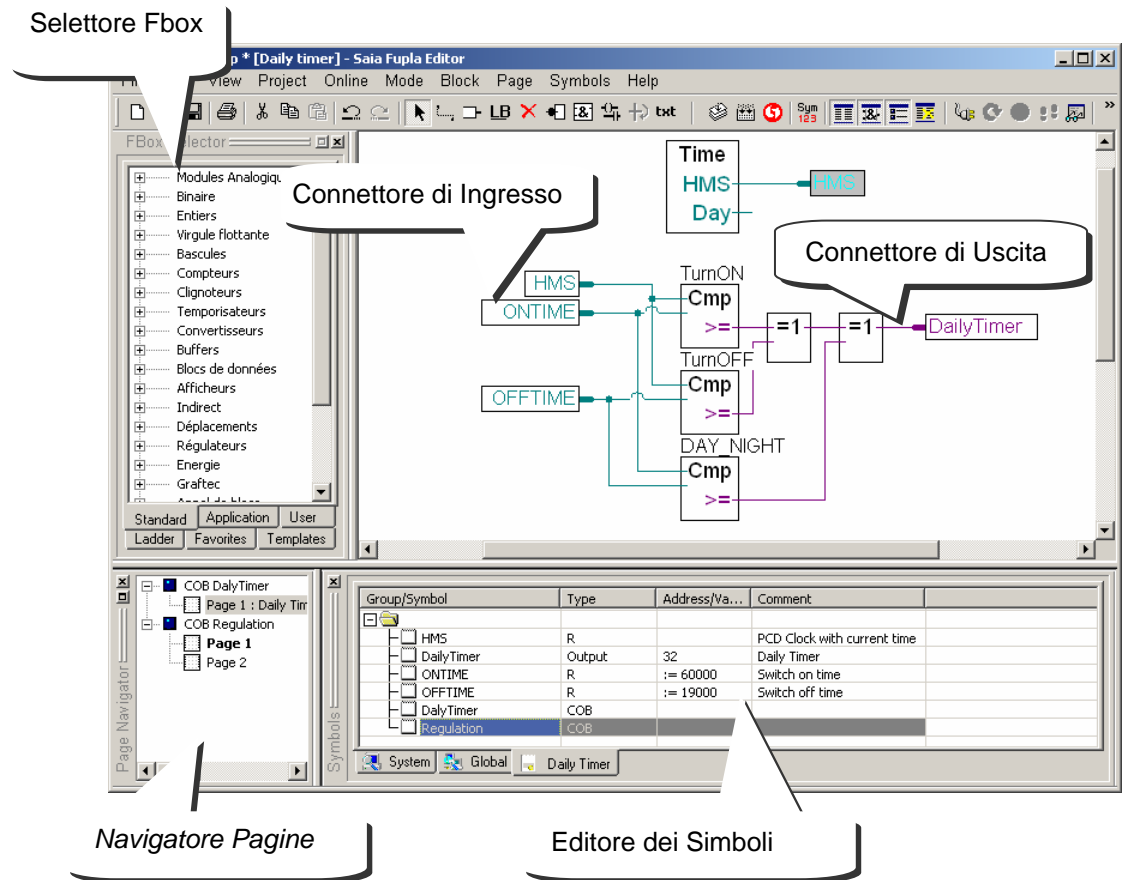
Per creare un nuovo file di programma all'interno di questo progetto, fare clic sul pulsante *Nuovo File* oppure utilizzare il pulsante destro del mouse:



Nuovo File



### 4.3 Organizzazione di una finestra Fupla



Il PCD legge le informazioni rappresentate dai connettori di Ingresso, le valuta in base al programma e scrive i risultati nei simboli associati ai connettori di Uscita. I simboli utilizzati dal programma sono tutti elencati nella finestra *Symbols (Simboli)*. Per i connettori di Ingresso e Uscita sono ammessi tutti i simboli, ad eccezione dei simboli di tipo *input (ingresso)* e *constant (costante)* digitali. Gli ingressi digitali e le costanti digitali prevedono dei dati di sola lettura e quindi possono essere usati solo in associazione a connettori di Ingresso.

Al centro della pagina è rappresentato il programma, costituito dalle varie funzioni grafiche selezionate dalla finestra *Selettore FBox*. I collegamenti rappresentano lo scambio dei dati tra le varie funzioni. Il colore di questi collegamenti definisce il tipo di dati: porpora per i dati binari (Booleani), blu per i numeri interi e giallo per i numeri in virgola mobile. I dati di tipo o colore diverso non possono essere collegati tra di loro senza essere stati prima convertiti in un tipo comune (*FBox: Standard, Converter – Standard, Convertitore*).

Se il programma utilizza diverse pagine, si può utilizzare la finestra *Navigatore Pagine* che permette, all'occorrenza, di cancellare alcune pagine ed aiuta a spostarsi rapidamente all'interno della struttura del programma.



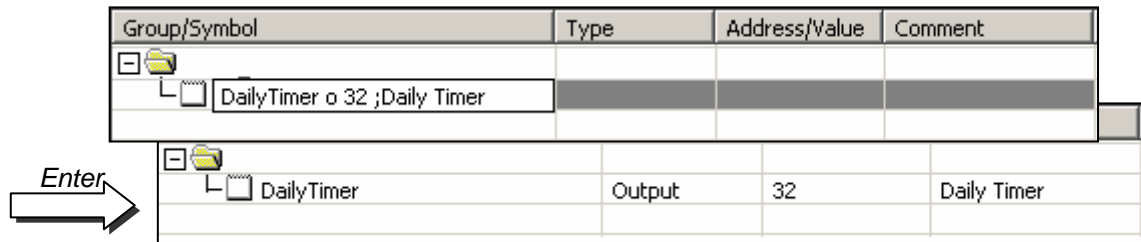


### 4.4.1 Come aggiungere un nuovo Simbolo nella finestra Simboli

#### Metodo semplice

Per aggiungere un simbolo all'elenco aprire la finestra Symbols (*Simboli*), posizionare il mouse al centro della finestra e fare click con il pulsante destro per selezionare il menu contestuale *Insert Symbol (Inserisci Simbolo)*. A questo punto compilare i campi: *Group/Symbol, Type, Address/Value e Comment (Gruppo/Simbolo, Tipo, Indirizzo/Valore e Commento)*.

#### Metodo rapido 1

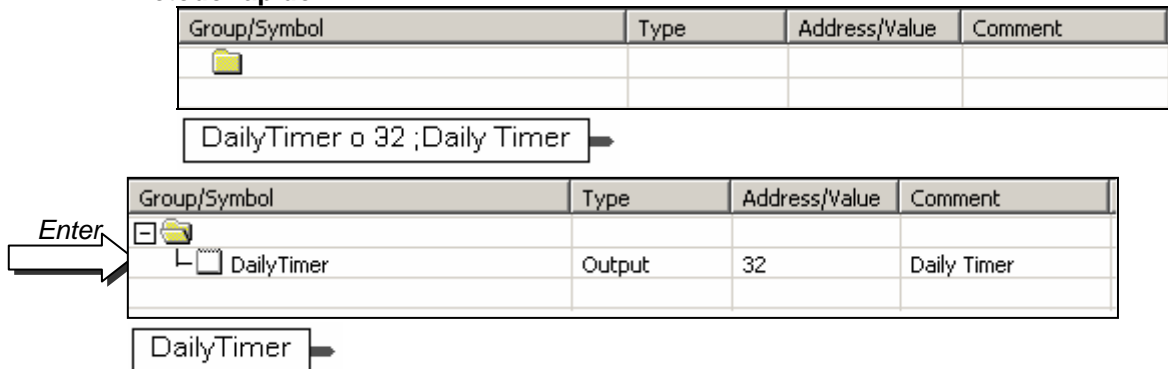


Sintassi da seguire:

nome\_simbolico tipo indirizzo ;commento

Se il nuovo simbolo è stato definito utilizzando la sintassi sopra indicata, le informazioni verranno inserite nei campi corretti semplicemente premendo il tasto Enter sulla tastiera.

#### Metodo rapido 2



Durante la fase di scrittura del programma si possono anche aggiungere nuovi connettori. Per fare questo, scrivere una riga di codice programma contenente il codice mnemonico e, per l'operando, inserire il nome simbolico e la definizione utilizzando la seguente sintassi:



nome\_simbolico tipo indirizzo ;commento

Il nuovo simbolo verrà automaticamente inserito nell'elenco della finestra *Symbols*, premendo il tasto *enter* sulla tastiera; in questo caso solo se la definizione del simbolo è corretta, e solo se è stata selezionata l'opzione *Automatically add entered type/value to the Symbol Table (Aggiunta automatica del tipo/valore inserito nella Tabella Simboli)* (menu *Tools, Options - Strumenti, Opzioni* nell'editor IL).

## 4.4.2 Modalità di indirizzamento dei simboli



La definizione di un simbolo non include necessariamente tutte le informazioni sotto indicate. Possiamo distinguere fra tre tipi di indirizzamento:

### Indirizzi assoluti

Group/Symbol	Type	Address/Value	Comment
 	Output	32	Daily Timer



I dati sono definiti solo con tipo e indirizzo (es. 32), ed un commento opzionale. L'utilizzo dell'indirizzamento assoluto direttamente nel programma si traduce in uno svantaggio quando si modifica il tipo o l'indirizzo. Il programma utente non verrà aggiornato dalle modifiche effettuate nell'elenco dei simboli. Le modifiche devono essere effettuate manualmente per ogni connettore del programma. E' quindi preferibile utilizzare nomi simbolici, con indirizzamento dinamico opzionale.

### Nomi simbolici

Group/Symbol	Type	Address/Value	Comment
  DailyTimer	Output	32	Daily Timer

I dati sono definiti con un nome simbolico, tipo, indirizzo e commento opzionale. La correzione di un simbolo, tipo o indirizzo è supportata dall'elenco dei simboli ed ogni connettore del programma utente viene automaticamente aggiornato al variare del simbolo.

### Indirizzamento dinamico

Group/Symbol	Type	Address/Value	Comment
  HMS	R		PCD Clock with

Si tratta di una forma di indirizzamento simbolico in cui l'indirizzo non è definito. L'indirizzo viene assegnato automaticamente durante la costruzione del programma. L'indirizzo viene acquisito in un campo di indirizzi definiti in *Software Settings (Impostazioni Software)*. (Vedere Project Manager.)

N.B.: L'indirizzamento dinamico è disponibile per flag, contatori, temporizzatori, registri, testi, DB, COB, PB, FB e SB. Per ingressi, uscite e XOB è necessario invece definire sempre indirizzi assoluti.

### 4.4.3 Utilizzo di un Simbolo dell'elenco Simboli in un programma FUPLA

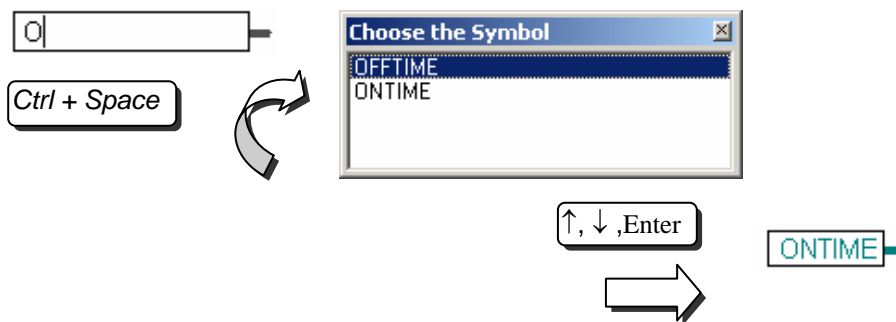
Durante la scrittura di un programma, possono essere utilizzati in vari modi i simboli già definiti nella finestra *Symbols (Simboli)*:

#### Inserimento di un simbolo da tastiera

Il nome del simbolo viene inserito interamente da tastiera per ogni istruzione che ne fa uso. Questo metodo potrebbe comportare degli errori di digitazione durante la fase di scrittura del nome simbolico, e questo verrebbe evidenziato solo all'atto della costruzione del programma.

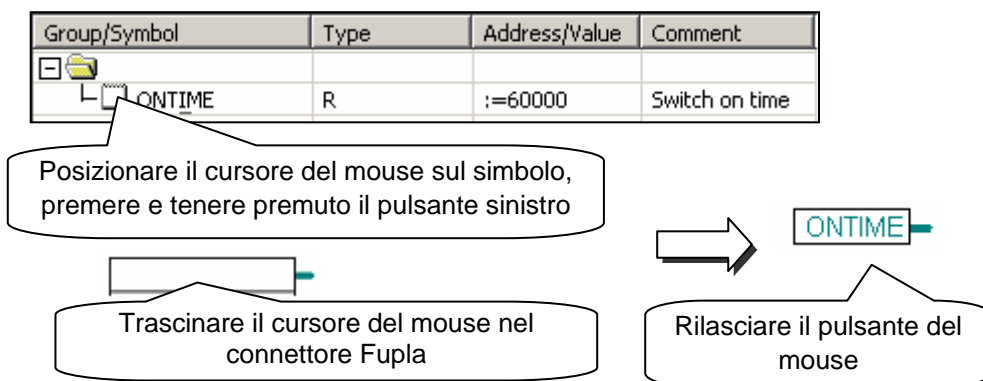
#### Inserimento di un simbolo tramite ricerca selettiva

Group/Symbol	Type	Address/Value	Comment
HMS	R		PCD Clock with current time
DailyTimer	Output	32	Daily Timer
ONTIME	R	:=60000	Switch on time
OFFTIME	R	:=19000	Switch off time



Se, dopo aver digitato solo alcuni caratteri del nome simbolico da tastiera si premono i tasti *Ctrl+Barra spaziatrice*, compare una finestra contenente l'elenco di tutti i simboli che iniziano con le lettere digitate. Il simbolo richiesto può quindi essere selezionato con il mouse o con i tasti freccia della tastiera (↑, ↓) e confermato premendo *Enter*.

#### Inserimento di un simbolo con il metodo “trascina e rilascia”

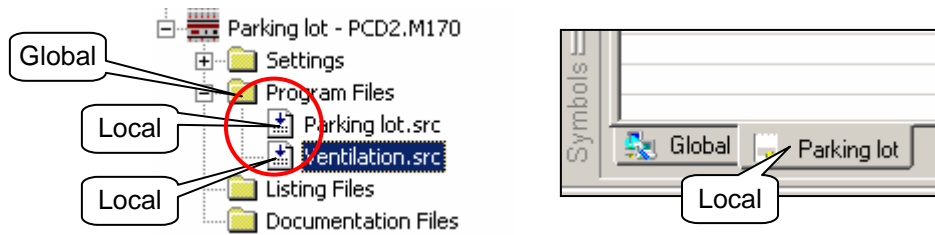


Questo metodo di utilizzo del simbolo esclude qualsiasi possibilità di errori di battitura. Nella finestra *Symbols (Simboli)*, posizionare il cursore del mouse sulla riga di definizione di un simbolo, quindi premere e tenere premuto il pulsante sinistro del mouse. Trascinare il cursore del mouse sopra un connettore vuoto e rilasciare il pulsante del mouse. Il simbolo scelto verrà automaticamente aggiunto nel connettore su cui è posto il cursore del mouse.

E' possibile anche trascinare il simbolo in un'area vuota della pagina Fupla: in questo modo, sarà possibile aggiungere automaticamente e con una sola operazione sia il connettore che il simbolo.

### 4.4.4 Simboli locali e globali

La finestra di definizione simboli contiene due cartelle: *Globale* e *Locale*



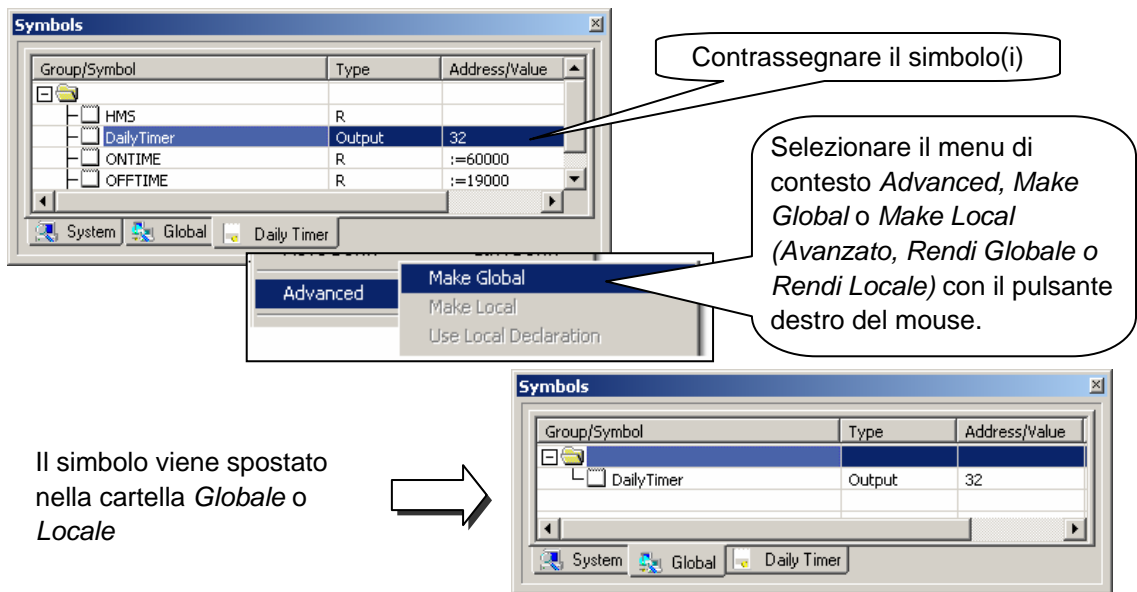
#### Definizione

I simboli locali compaiono in una cartella contenente il nome del file che li utilizza. Questi simboli possono essere utilizzati esclusivamente all'interno di quel file. (*Parking lot.src*)

I simboli Globali che compaiono nella cartella *Globale* possono essere utilizzati da tutti i file all'interno della CPU (*Parking lot.src* e *Ventilation.src*)

#### Conversione Locale/Globale

Se necessario, i simboli della finestra *Symbols (Simboli)* possono essere spostati dalla cartella locale alla cartella globale e viceversa.



Qualsiasi nuovo simbolo definito direttamente dall'ambiente editore Fupla verrà aggiunto nella cartella Globale o nella cartella Locale, in base alle impostazioni definite mediante l'opzione *Add symbols to Global table (Aggiungi Simboli Globali)*. Vedere il menu contestuale *Advanced, Options (Avanzate, Opzioni)* nella finestra *Symbols (Simboli)*.

## 4.5 Editare i connettori

I connettori di Ingresso ed Uscita possono essere posizionati ovunque all'interno delle pagine Fupla ed utilizzati per contenere i simboli necessari alle funzioni di programmazione descritte dagli FBox.

Per default, su ogni nuova pagina creata viene predisposta una serie di connettori di ingresso (a sinistra) e di uscita (a destra). Se non si desidera che le nuove pagine contengano le suddette serie di connettori ma si preferisce posizionarli manualmente in base alle specifiche esigenze, disattivare la relativa opzione selezionando la voce di menu: *View, Options..., Layout, New pages with side connectors (Visualizza, Opzioni..., Imposta Pagina, Nuove pagine con connettori laterali)*.

Per rimuovere i connettori vuoti presenti sulla sinistra o sulla destra della pagina, selezionare la voce di menu: *Page, Remove Empty connectors (Pagina, Rimuovi Connettori Vuoti)*.

Per reinserire nuovamente le serie di connettori vuoti all'interno di una pagina vuota, selezionare la voce di menu: *Page, Add Empty Side Connectors (Pagina, Inserisci Connettori Laterali Vuoti)*.

### 4.5.1 Inserire i connettori



#### *Inserisci Connettori*

Per aggiungere un connettore ed il relativo simbolo (risorsa) ad una pagina Fupla, selezionare il pulsante *Place Connector (Inserisci Connettore)* della barra degli strumenti quindi posizionare il cursore del mouse all'interno della pagina Fupla.

Per inserire un connettore di ingresso (lettura), è sufficiente fare poi clic con il tasto sinistro del mouse. Per inserire invece un connettore di uscita (scrittura), tenere premuto il tasto *Shift (Maiusc)* quindi fare clic con il tasto sinistro del mouse.

Il connettore appena inserito risulterà immediatamente predisposto alla definizione del relativo simbolo; al suo interno verrà infatti visualizzato il cursore di inserimento.

Se non si desidera definire immediatamente il simbolo, premere il tasto *ESC* e procedere all'inserimento del connettore successivo

### 4.5.2 Modificare il simbolo interno ad un connettore

Per definire o modificare il simbolo (risorsa) di un connettore già presente sulla pagina Fupla, è sufficiente selezionare il connettore interessato facendo doppio clic su di esso. All'interno del connettore verrà visualizzato il cursore di inserimento. A questo punto è possibile specificare il simbolo e la sua completa definizione.

Notare che i nuovi simboli inseriti nei connettori vengono automaticamente aggiunti all'elenco dei simboli visualizzato all'interno della finestra *Symbols (Simboli)*.

### 4.5.3 Inserire in modo rapido un simbolo e il connettore associato

I simboli già presenti all'interno della finestra *Symbols (Simboli)* possono essere trascinati direttamente in una zona vuota della pagina Fupla: verrà creato automaticamente un nuovo connettore contenente il simbolo selezionato.

Se il simbolo viene trascinato sull'ingresso o sull'uscita di un FBox, a quest'ultimo verrà collegato direttamente un connettore di ingresso o di uscita.

#### 4.5.4 Trascinare, Copiare/Incollare e Cancellare un simbolo



Selezionando l'area di colore rosso, le successive operazioni influenzeranno il solo simbolo. Dopo aver selezionato il simbolo con il mouse, sarà possibile trascinarlo o copiarlo/incollarlo su un altro connettore oppure cancellarlo. Facendo clic con il tasto destro del mouse verrà visualizzato un menu contestuale contenente tutte le operazioni eseguibili.

#### 4.5.5 Copiare/Incollare e Cancellare un connettore



Selezionando l'area di colore bianco, le successive operazioni influenzeranno sia il connettore che il simbolo in esso contenuto. Facendo clic con il tasto destro del mouse verrà visualizzato un menu contestuale contenente tutte le operazioni eseguibili.

#### 4.5.6 Estendere i connettori

I connettori sono "estendibili". Ciò significa che è possibile definire il numero di connettori con un movimento verticale del mouse.

Selezionare il pulsante: Select Mode (*Modo Selezione*)

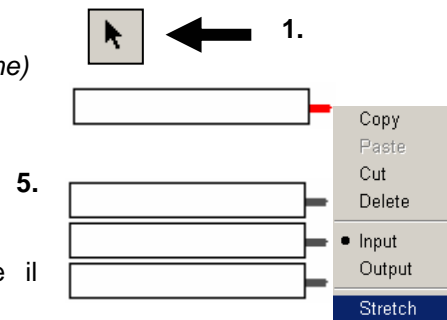
Selezionare l'area di colore rosso del connettore.

Visualizzare il menu contestuale facendo clic con il tasto destro del mouse.

Selezionare la voce di menu: *Stretch (Estendi)*

Spostare verticalmente il mouse fino a definire il numero di connettori desiderato

Premere il tasto sinistro del mouse.



#### 4.5.7 Spostare verticalmente un connettore

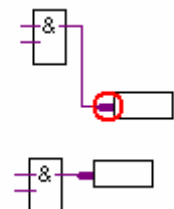
Per spostare il connettore, posizionare il cursore del mouse nell'area indicata dal cerchietto di colore rosso.

Premere e tenere premuto il tasto *Shift (Maiusc)*.

Premere e tenere premuto il tasto sinistro del mouse.

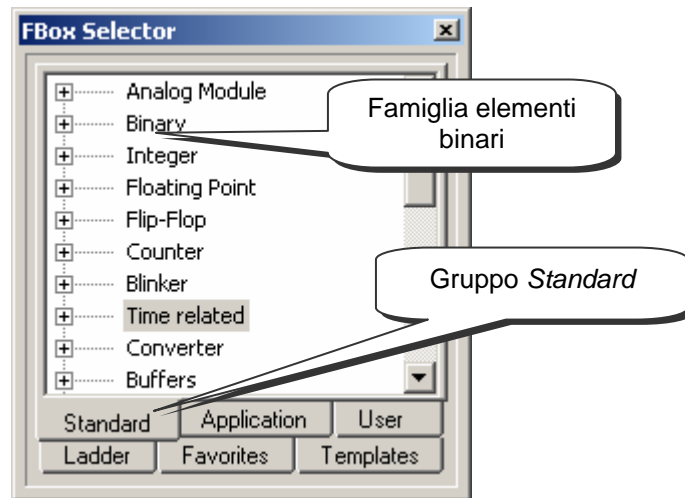
Trascinare verticalmente il mouse verso una zona vuota della pagina.

Rilasciare il tasto del mouse ed il tasto Shift.



## 4.6 Scrittura di una funzione Fupla

### 4.6.1 Selettore FBox



Aggiungi  
FBox

Tutte le funzioni grafiche (FBox) necessarie per produrre programmi sono elencate nella finestra selettore FBox. Tali funzioni sono suddivise in gruppi.

Il gruppo *Standard* comprende tutte le funzioni di base necessarie per tutti gli utenti.

Il gruppo *Application* comprende le funzioni specifiche per determinati ambienti professionali specializzati.

Il gruppo *Ladder* comprende tutte le funzioni richieste per la programmazione ladder, sotto forma di schemi a contatti.

Ciascun gruppo è a sua volta suddiviso in famiglie contenenti tutte le funzioni che coprono un particolare campo di applicazione. Ad esempio, il gruppo *Standard* comprende le seguenti famiglie di funzioni:

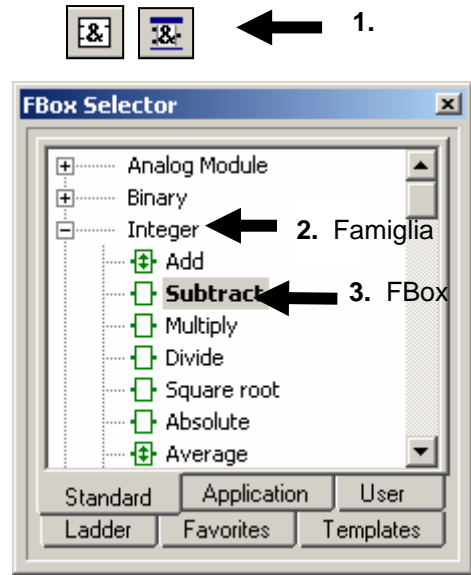
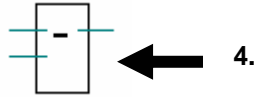
<i>Binary</i>	FBox per produrre equazioni logiche
<i>Integer</i>	FBox per operazioni aritmetiche con numeri interi
<i>Floating Point</i>	FBox per operazioni aritmetiche in virgola mobile
<i>Counter</i>	FBox per operazioni di conteggio
<i>Time related</i>	FBox per operazioni di temporizzazione
<i>Analogue Module</i>	FBox per la gestione di moduli analogici
<i>Communication</i>	FBox per lo scambio dati: registri, flag, ... su reti S-Bus o Ethernet
<i>Converter</i>	FBox per la conversione da binario a intero, da intero a virgola mobile ...

...

### 4.6.2 Scrittura FBox

Le funzioni necessarie per scrivere un programma devono essere selezionate dalla finestra *Selettore Fbox (Selettore FBox)*, ed inserite nel programma Fupla.

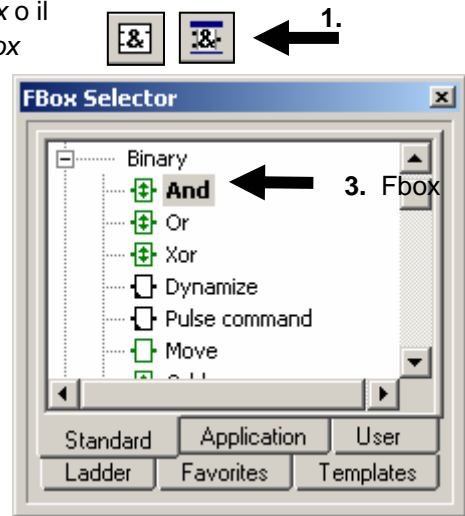
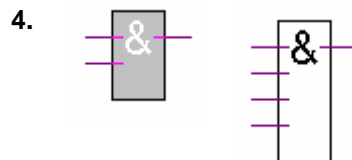
1. Selezionare il pulsante *Add (Aggiungi) FBox* o il selettore *Show/Hide (Mostra/Nascondi) FBox..*
2. Aprire una famiglia di funzioni.
3. Selezionare una funzione.
4. Posizionare la funzione nella pagina che si sta scrivendo, quindi premere il pulsante sinistro del mouse.



### 4.6.3 Scrittura di Fbox estendibili

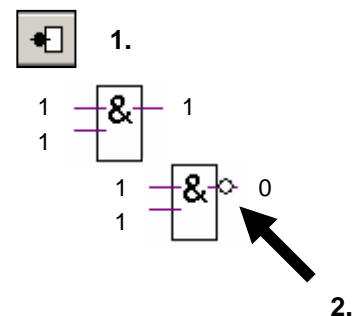
Alcuni FBox sono estendibili, nel senso che il numero di collegamenti può essere definito muovendo verticalmente il mouse.

1. Selezionare il pulsante *Add (Aggiungi) FBox* o il selettore *Show/Hide (Mostra/Nascondi) FBox*
2. Aprire una famiglia di funzioni.
3. Selezionare una funzione.
4. Posizionare la funzione nella pagina che si sta scrivendo, quindi premere il pulsante sinistro del mouse.
5. Muovere il mouse verticalmente per definire il numero di ingressi.
6. Premere il pulsante sinistro del mouse.



### 4.6.4 Inversione logica

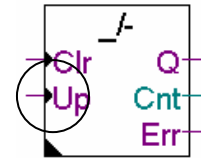
1. Selezionare il pulsante *Inverti Connessione Binaria*.
2. Posizionare il puntatore del mouse sul collegamento di ingresso o di uscita di una funzione logica e premere il pulsante sinistro del mouse.





### 4.6.5 Dinamizzazione

Gli ingressi di alcune funzioni binarie sono stati “dinamizzati”. Questo significa che prendono in considerazione solo il fronte positivo di un segnale logico. Queste funzioni sono identificate da un piccolo triangolo nero.



Ad esempio, un contatore di impulsi non può essere incrementato quando il suo ingresso *UP* è alto.

Fbox: *Counter, Up with clear (Contatore, Avanti con cancellazione)*

Altrimenti, cosa succederebbe se il segnale *UP* rimanesse alto per un determinato periodo di tempo? Il contatore continuerebbe ad incrementarsi fin tanto che il segnale *UP* rimane alto. È appunto per questo tipo di applicazione che certi ingressi digitali sono stati “dinamizzati”. In questo modo, solo il fronte positivo del segnale *UP* farà incrementare il contatore.

Talvolta è necessario aggiungere la dinamizzazione all'ingresso o all'uscita di un FBox. Per fare questo è sufficiente utilizzare la funzione *Bynary, Dynamize (Binaria, Dinamizza)*.



### 4.6.6 Commenti

Nel programma si possono inserire dei commenti:

1. Selezionare il pulsante *Inserisci Commento*
2. Posizionare il commento nella pagina del programma e premere il pulsante sinistro del mouse.
3. Scrivere il commento.
4. Premere il pulsante *ENTER*.



Daily Timer

### 4.6.7 Aiuto sugli FBox

Per ottenere la descrizione completa di una qualsiasi funzione, selezionare il relativo FBox nella finestra *Fbox Selector (Selettore FBox)* e premere il tasto F1.

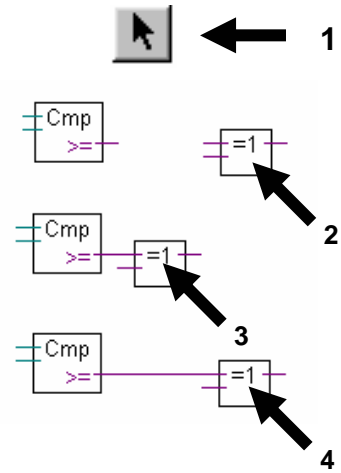
In alternativa, è possibile posizionare il puntatore del mouse sull'FBox desiderato nel programma e fare doppio clic col pulsante sinistro del mouse.

Per una rapida identificazione di un FBox sconosciuto all'interno di un programma, richiamare la finestra *Fbox Selector (Selettore FBox)*, posizionare il puntatore del mouse sull'FBox sconosciuto e fare clic con il pulsante sinistro del mouse. La finestra *Selettore FBox* visualizzerà a questo punto la funzione selezionata nel programma.

## 4.7 Collegare FBox e connettori

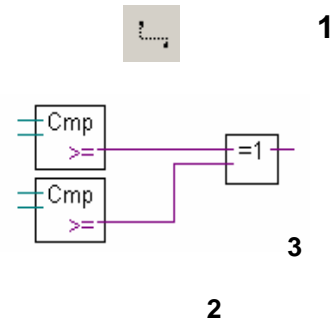
### 4.7.1 Collegamento mediante trascinamento dell'FBox

1. Fare clic sul pulsante *Selezione* nella barra degli strumenti.
2. Puntare il mouse sull'Fbox desiderato, e premere il pulsante sinistro del mouse.
3. Tenere premuto il pulsante del mouse e trascinare l'FBox in orizzontale verso un altro Fbox situato nelle vicinanze.
4. I due FBox vengono collegati non appena i due punti di connessione si toccano.



### 4.7.2 Collegamento con instradamento automatico

1. Fare clic sul pulsante *Auto Line Mode (Modo Linee Automatiche)* della barra degli strumenti.
2. Posizionare il cursore del mouse sul punto di partenza del collegamento quindi fare clic con il tasto sinistro.
3. Posizionare il cursore del mouse sul punto di fine collegamento quindi fare nuovamente clic con il tasto sinistro.



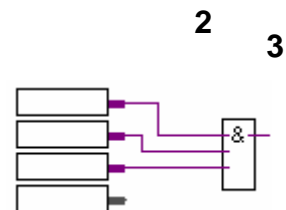
Nota:

E' possibile selezionare anche punti intermedi del percorso.

Per interrompere la generazione del collegamento, premere il tasto destro del mouse.

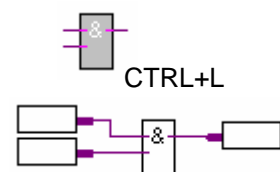
### 4.7.3 Collegamento multiplo con instradamento automatico

1. Selezionare la voce di menu *Mode, Connect Bus (Modo, Bus di Collegamento)* oppure premere la combinazione di tasti (CTRL+B).
2. Selezionare il punto di partenza con il mouse.
3. Selezionare quindi il punto finale.



### 4.7.4 Collegare tutti gli Ingressi/Uscite di un FBox a connettori

Posizionare il cursore del mouse sopra un FBox. Fare clic con il tasto destro del mouse per visualizzare il menu contestuale e selezionare la voce: *Connections, Connect All (Collegamenti, Collega Tutto)*.



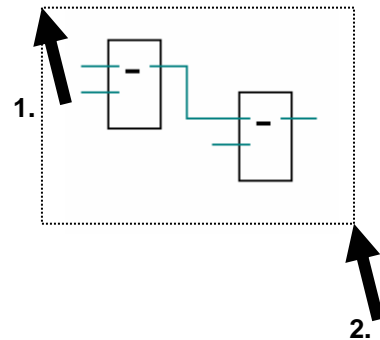
### 4.7.5 Cancellare linee, FBox, connettori o simboli

Selezionare il pulsante *Modo Cancellazione* della barra degli strumenti quindi selezionare le linee di collegamento, gli FBox, i connettori o i simboli che si desidera cancellare.



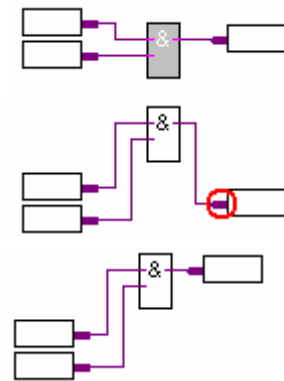
Un'altra soluzione più rapida è quella di selezionare l'area interessata e cancellare gli elementi in essa contenuti.

- 1 Premere il tasto sinistro del mouse.
- 2 Senza rilasciare il tasto, trascinare il mouse fino a selezionare l'area interessata.
- 3 Rilasciare il tasto del mouse.
- 4 Selezionare la voce di menu *Edit, Delete* (*Modifica, Cancella*)



### 4.7.6 Spostare verticalmente FBox/connettori senza rimuovere i collegamenti

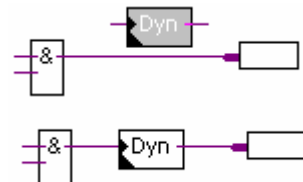
- 1 Posizionare il cursore del mouse sopra l'FBox.
- 2 Premere e tenere premuto il tasto Shift (Maiusc).
- 3 Premere e tenere premuto il tasto sinistro del mouse.
- 4 Trascinare verticalmente il mouse verso un'area vuota della pagina.
- 5 Rilasciare il tasto del mouse e il tasto *Shift* (Maiusc).



Per spostare il connettore, posizionare il cursore del mouse nella zona indicata dal cerchietto rosso quindi ripetere la suddetta sequenza.

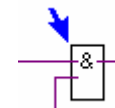
### 4.7.7 Inserire FBox senza rimuovere i collegamenti

- 1 Selezionare l'FBox che si desidera inserire all'interno del Selettore FBox.
- 2 Posizionare l'FBox interessato sopra il collegamento.



### 4.7.8 Regole da osservare

Non sono ammessi "collegamenti di richiusura" (loop). In caso contrario, verrà generato il messaggio di errore: *Page 1: Error 55: Loop back detected (Pagina 1: Errore 55: Rilevata Richiusura)*



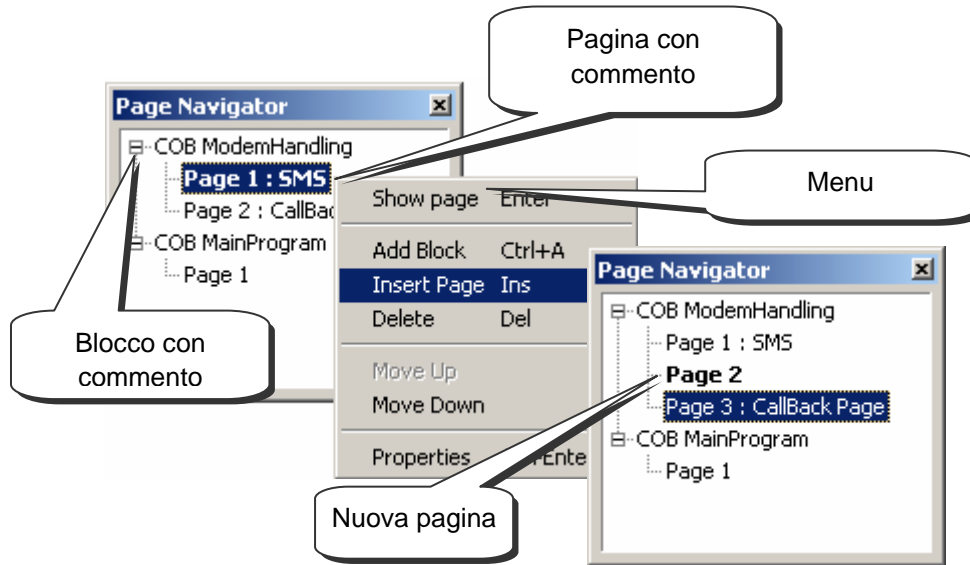
Non sono ammessi collegamenti diretti tra connettori di ingresso e connettori di uscita, ma è necessario interporre un FBox di tipo: *Binary, Direct transfer (Binario, Trasferimento Diretto)* o *Integer, Direct transfer (Intero, Trasferimento Diretto)*.



I simboli (risorse) di un connettore di uscita devono sempre essere collegati ad un FBox. In caso contrario, verrà generato il messaggio di errore: *Page 1: Error 53: Incomplete network (Pagina 1: Errore 53: Rete Incompleta)*



## 4.8 Scrittura di pagine Fupla



Mostra/Nascondi  
Navigatore  
Pagine

La finestra *Page Navigator* (*Navigatore Pagine*) visualizza i blocchi e le pagine del programma. Ogni file Fupla può contenere fino a 200 pagine raggruppate in blocchi: COB, PB, FB, o SB. Il programma Fupla risulta tuttavia più veloce se non vi sono troppe pagine in un singolo file. Per default, le pagine vengono inserite in un blocco di tipo COB. Per informazioni più dettagliate sui blocchi e sull'utilizzo dei medesimi, si prega di fare riferimento al capitolo 5 del presente manuale.

### 4.8.1 Inserimento di pagine



Inserisci Pagina

Aprire la finestra *Page Navigator* (*Navigatore Pagine*), contrassegnare la pagina di riferimento e selezionare *Insert Page* (*Inserisci Pagina*) dal menu.

E' possibile anche inserire una pagina dopo la pagina corrente utilizzando il pulsante *Insert* (*Inserisci*) oppure la voce di menu: *Page Insert After* o *Page Insert Before* (*Inserisci Pagina Dopo/Inserisci Pagina Prima*)

### 4.8.2 Cancellazione di una pagina

Aprire la finestra *Navigatore Pagine*, contrassegnare la pagina da cancellare e selezionare *Delete* (*Cancella*) dal menu.

### 4.8.3 Navigazione tra le pagine

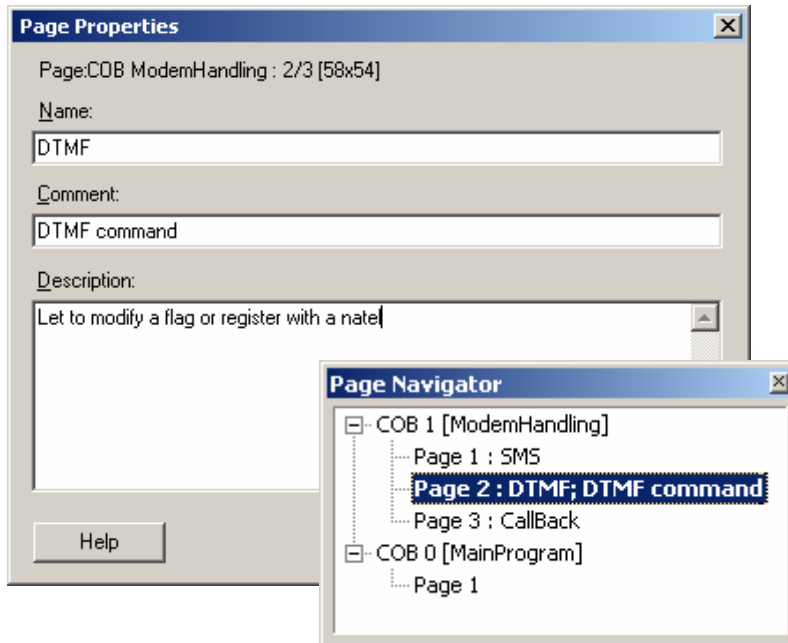


Pagina  
Precedente  
Successiva

E' possibile navigare tra le pagine anche utilizzando i pulsanti *Pagina Precedente* e *Pagina Successiva*, che permettono di spostarsi da una pagina all'altra di un blocco Fupla. Quando uno dei due pulsanti assume il colore grigio, significa che è stata raggiunta la prima o l'ultima pagina del blocco.

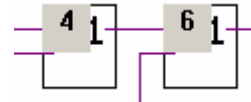
#### 4.8.4 Documentazione della pagina

Si consiglia vivamente di documentare ogni singola pagina di un programma Fupla. Questo risulterà estremamente utile durante la navigazione tra le varie pagine del programma, in quanto i nomi e i commenti delle pagine vengono visualizzati nella finestra *Page Navigator (Navigatore Pagina)*. La descrizione è un modo per fornire alcune utili informazioni sul programma, che ne renderanno più facile la manutenzione.



#### 4.8.5 Elaborazione del programma da parte del PCD

Il PCD elabora le pagine di ciascun blocco iniziando dalla parte superiore sinistra della prima pagina fino alla parte inferiore destra dell'ultima pagina. Per maggiori dettagli sull'ordine con cui il PCD elabora gli Fbox, selezionare in sequenza le voci di menu: *Page, FBox priorities (Pagina, Priorità FBox)*.

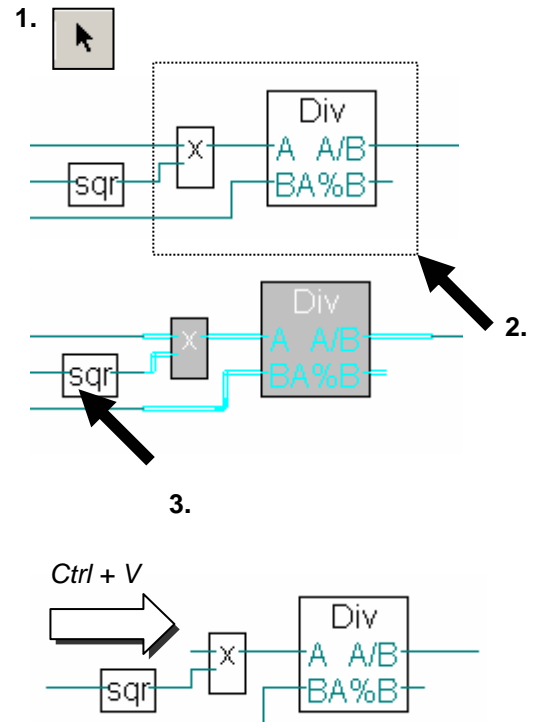


## 4.9 Copia e incolla

Alcune parti di un programma possono essere ripetitive. Non è quindi necessario riscriverle completamente. Può risultare molto più veloce duplicarle utilizzando la funzione copia e incolla, ed adattarle in base alle esigenze.

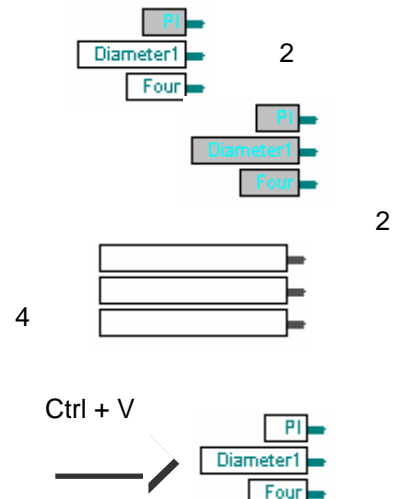
### 4.9.1 Come copiare e incollare una parte di programma

1. Fare clic sul pulsante *Seleziona*.
2. Selezionare l'area da copiare:
  - Premere il tasto sinistro del mouse.
  - Trascinare il mouse tenendo premuto il tasto.
  - Rilasciare il tasto del mouse.
3. Aggiungere un FBox o un collegamento all'area selezionata:
  - Premere il tasto Ctrl.
  - Tenendo premuto il tasto Ctrl selezionare i connettori e gli FBox da aggiungere.
4. Copiare l'area selezionata utilizzando le voci di menu *Edit, Copy (Modifica, Copia)*, oppure utilizzando i tasti *Ctrl + C*.
5. Incollare una copia dell'area selezionata utilizzando in sequenza *Edit Paste (Modifica, Incolla)* oppure i tasti *Ctrl + V*.
6. Posizionare la copia nella pagina Fupla:
  - Posizionare il puntatore del mouse al centro della copia.
  - Premere il tasto sinistro del mouse.
  - Trascinare il mouse tenendo premuto il tasto.



### 4.9.2 Come copiare e incollare simboli

1. Fare clic sul pulsante *Seleziona*.
2. Contrassegnare un elenco di simboli:
  - Posizionare il puntatore del mouse sul primo simbolo.
  - Fare clic con il tasto sinistro del mouse.
  - Posizionare il puntatore del mouse sull'ultimo simbolo.
  - Premere il tasto Shift. \*)
  - Tenendo premuto il tasto *Shift* fare clic con il tasto sinistro del mouse.
3. Copiare i simboli selezionati utilizzando in sequenza le voci di menu *Edit, Copy (Modifica, Copia)*, oppure i tasti *Ctrl + C*.
4. Posizionare il puntatore del mouse in una parte libera del margine.
5. Incollare la copia dei simboli selezionati utilizzando in sequenza le voci di menu *Edit, Paste (Modifica, Incolla)* oppure i tasti *Ctrl + V*.



\*) Il tasto *Ctrl* permette di selezionare simboli non consecutivi.

## 4.10 Esportazione e importazione di pagine

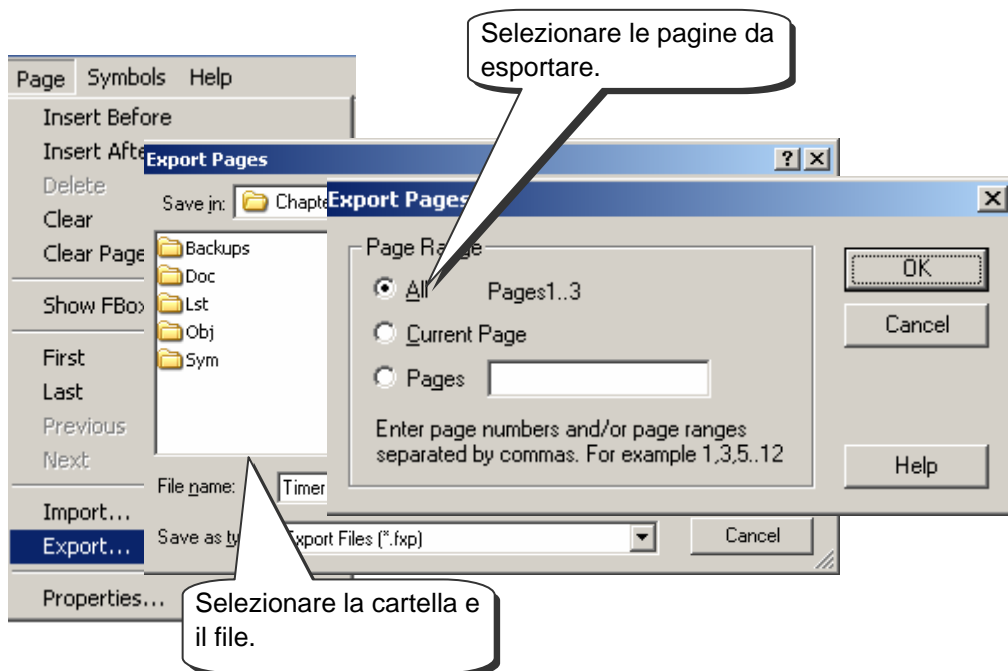
L'esportazione/importazione di pagine è uno strumento complementare alla funzione taglia/incolla, ma molto più potente. In particolare offre i seguenti vantaggi:

- Possibilità di copiare e incollare più pagine
- Possibilità di modificare simboli ed indirizzi importati
- Possibilità di creare una libreria di componenti

### 4.10.1 Esportazione pagina

La funzione di esportazione pagina può essere utilizzata per selezionare una o più pagine da un programma Fupla e salvarle all'interno di un file – insieme ai relativi FBox, collegamenti, commenti e simboli. In questo modo, parti di programmi attualmente utilizzati in uno o più file (\*.fxp) possono essere raggruppate per formare una libreria di componenti, utilizzabile per costruire più rapidamente applicazioni future.

Ad esempio, supponiamo che si debbano utilizzare frequentemente dei temporizzatori giornalieri, settimanali o mensili per attivare (on) o per disattivare (off) un'uscita digitale ad un'ora specifica di un qualsiasi giorno, settimana o mese. Per fare questo si può creare un programma di diverse pagine, in cui ogni pagina corrisponde ad uno dei temporizzatori attualmente utilizzati, ed esportare il tutto su un file: *Timer.fxp*.

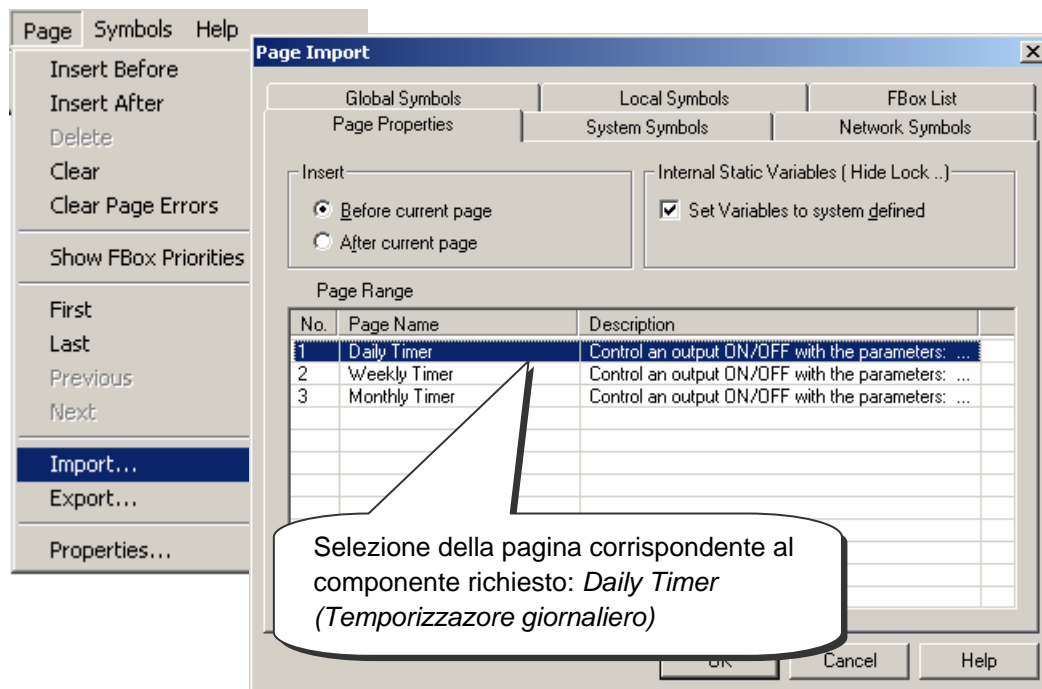


### 4.10.2 Importazione pagina

La funzione di importazione pagina supporta la selezione di una o più pagine da un file (\*.fxp) e la modifica dei simboli importati nel programma Fupla.

La selezione della pagina (o delle pagine) da importare può essere effettuata selezionando in sequenza *Page, Import, Page Properties (Pagina, Importa, Proprietà Pagina)*.

Ritornando al nostro esempio, ogni qualvolta serve un temporizzatore, si può importare il file *Timer.fxp* e selezionare la pagina (o pagine) corrispondenti al componente richiesto per l'applicazione desiderata.



Le finestre *Global Symbols (Simboli Globali)* e *Local Symbols (Simboli Locali)* visualizzano l'elenco dei simboli corrispondenti alle pagine selezionate. È comunque sempre possibile modificare il nome e il valore di ciascun simbolo.

Il modo più rapido per modificare i nomi di tutti i simboli in un programma importato, è quello di selezionare e raggruppare i simboli stessi.

Il menu *Insert Pre-group (Inserisci Pre-gruppo)* permette di inserire in un gruppo i simboli selezionati con un nome a scelta.

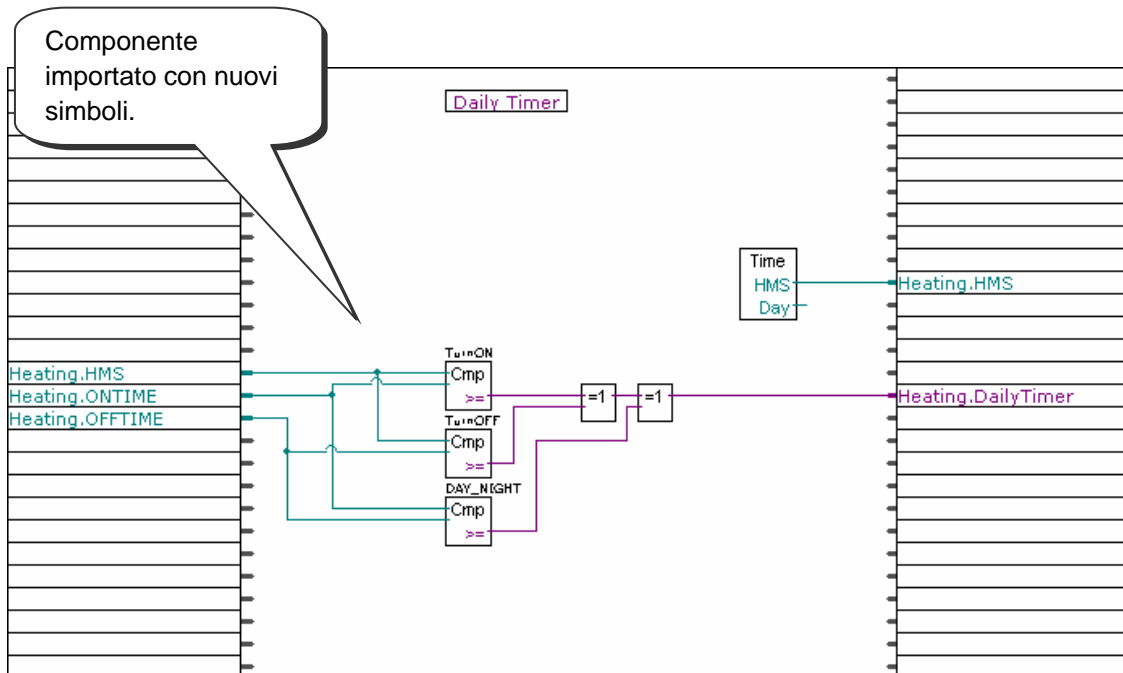
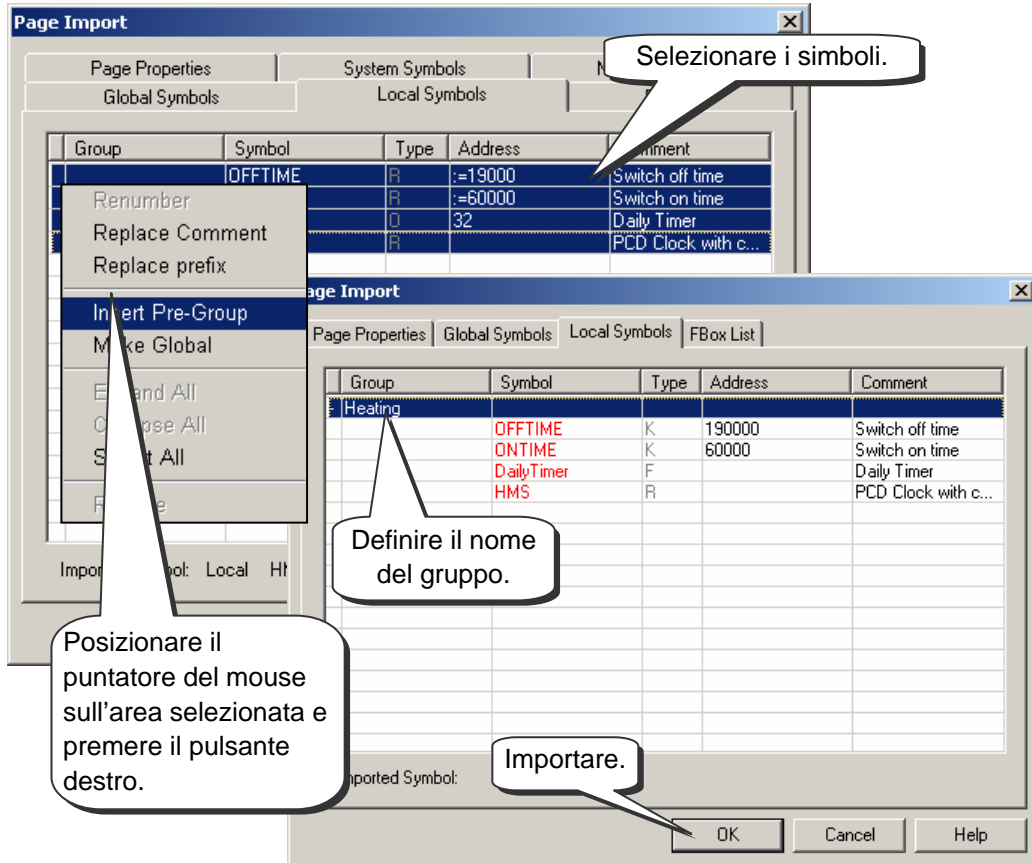
Analogamente, il modo più rapido per modificare gli indirizzi di tutti i simboli delle pagine importate è quello di selezionare e ri-numerare i simboli stessi.

Le voci del menu *Renumber (Ri-numerazione)* permette di modificare gli indirizzi dei simboli modificati, utilizzando un offset o un indirizzo base.

La finestra *FBox List (elenco FBox)* visualizza l'elenco dei simboli legati ad un gruppo di FBox. Questi simboli permettono di accedere alle informazioni interne dell'FBox, quali gli indirizzi dei parametri relativi alla finestra di modifica. Tale lista permette di modificare i nomi legati agli FBox ottenuti da pagine importate.



Ritornando al nostro esempio, proponiamo di importare i simboli dal componente “daily timer” in un gruppo denominato *Heating (Riscaldamento)*. Per ogni nuovo temporizzatore giornaliero, è opportuno definire un nome gruppo differente. In questo modo, i simboli dei componenti utilizzati più di una volta apparterranno a gruppi diversi e non comporteranno alcun problema di indirizzamento nella fase di costruzione del programma.



## 4.11 Scrittura del primo programma Fupla

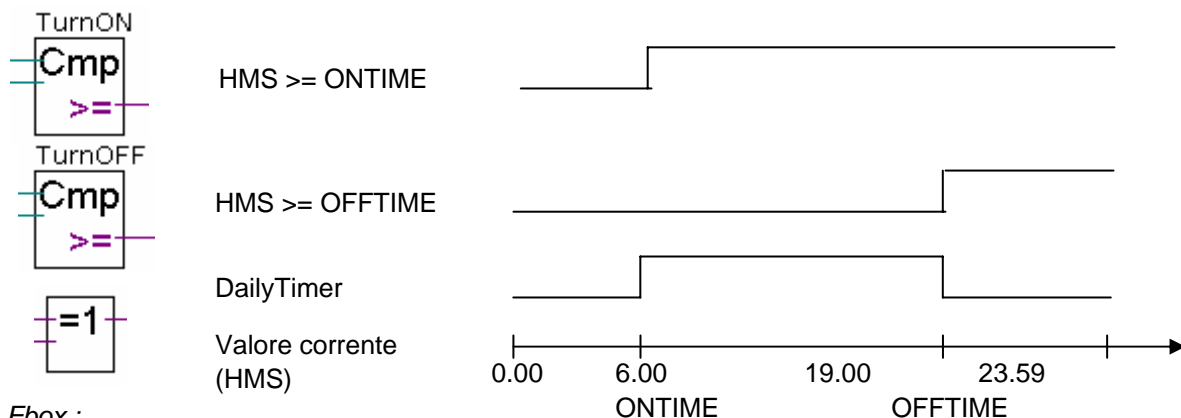
### 4.11.1 Obiettivo

Ora che l'ambiente di lavoro è noto, il passo successivo sarà quello di creare un programma più complesso rispetto alle strutture logiche presentate fino a questo punto. A questo proposito proponiamo di creare un temporizzatore giornaliero per comandare una uscita digitale (O 32) che si deve attivare (on) alle ore 06.00 e disattivare (off) alle ore 19.00. Sebbene questa funzione sia già disponibile nella libreria HEAVAC, proviamo a riprodurla da soli utilizzando degli Fbox standard.

### 4.11.2 Metodo

Prima di iniziare la programmazione, è necessario individuare un metodo, in linea con il nostro documento di specifiche, che possa essere implementato con le più elementari funzioni possibili.

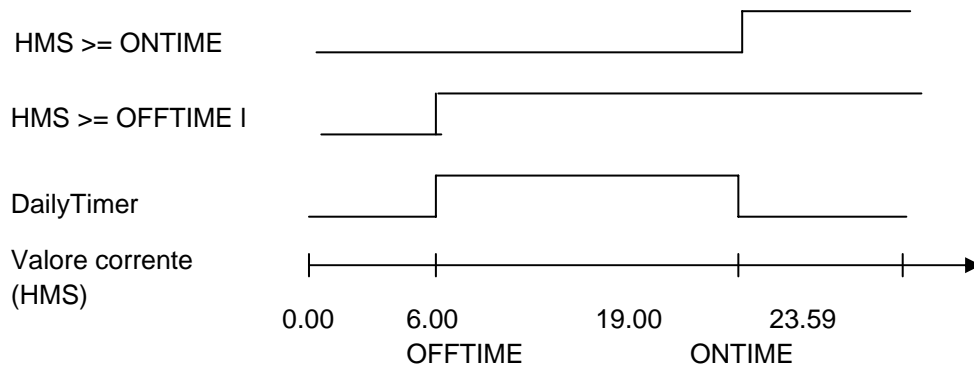
Per questo esempio di temporizzatore, proponiamo di effettuare due comparazioni. La prima consisterà nello stabilire se l'ora corrente in HMS (cioè l'ora indicata dai nostri orologi o l'ora del PCD) è superiore o uguale all'ora di accensione: *ONTIME*. Il secondo consisterà invece nello stabilire se l'ora corrente è inferiore o uguale all'ora di spegnimento: *OFFTIME*. Se entrambe le comparazioni sono verificate da una espressione – una funzione logica OR – l'uscita digitale 32 *DailyTimer* del temporizzatore deve essere attivata (on).



Fbox :

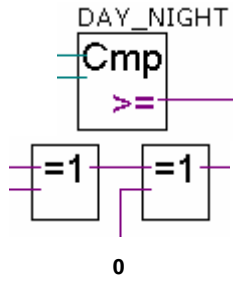
- Intero, Maggiore o uguale a
- Binario, Xor

Questo algoritmo offre una soluzione, ma lascia alcuni punti aperti. Cosa succede se le istruzioni di accensione e spegnimento si sovrappongono? Il diagramma seguente dimostra che l'uscita del PCD assumerà lo stato opposto a quello desiderato.

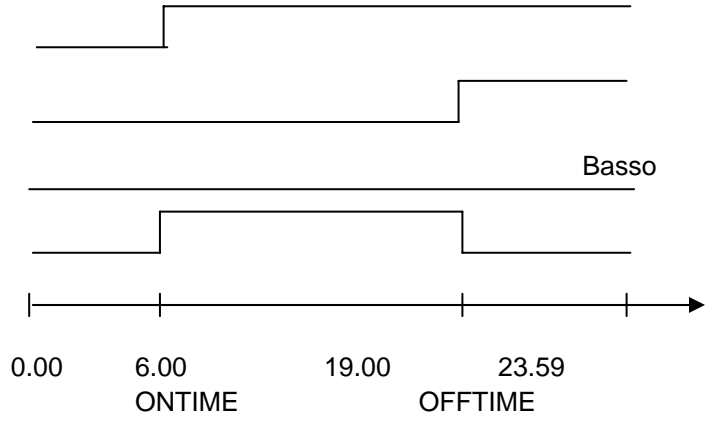


È necessario quindi completare il nostro algoritmo aggiungendo una terza comparazione per stabilire se l'ora di accensione è superiore o uguale all'ora di spegnimento. La soluzione finale sarà quindi la seguente.

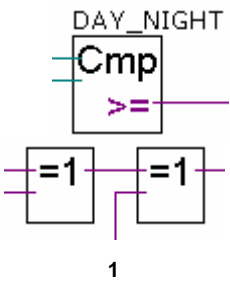
Uscite attive di giorno:



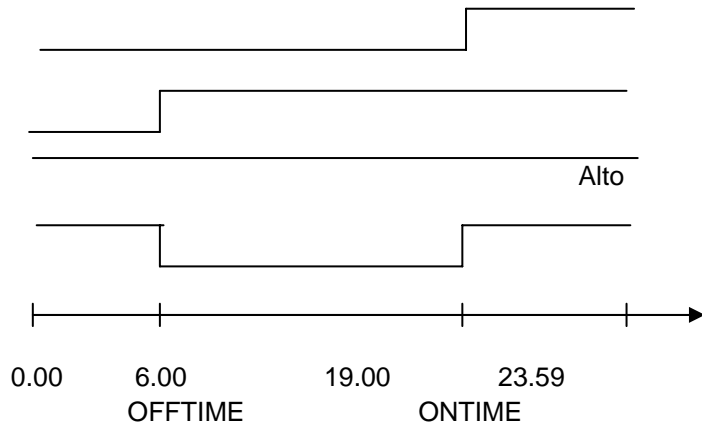
HMS >= ONTIME  
 HMS >= OFFTIME  
 ONTIME >= OFFTIME  
 DailyTimer  
 Valore corrente (HMS)



Uscite attive di notte:

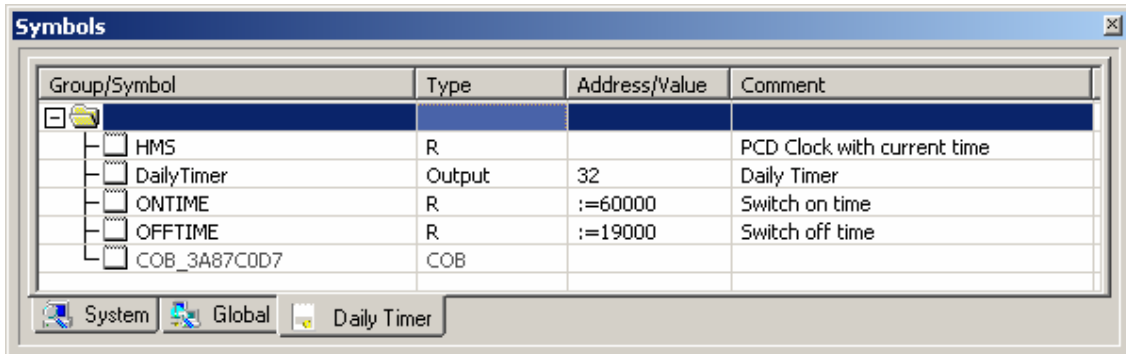


HMS >= ONTIME  
 HMS >= OFFTIME  
 ONTIME >= OFFTIME  
 DailyTimer  
 Valore corrente (HMS)



### 4.11.3 Programmazione

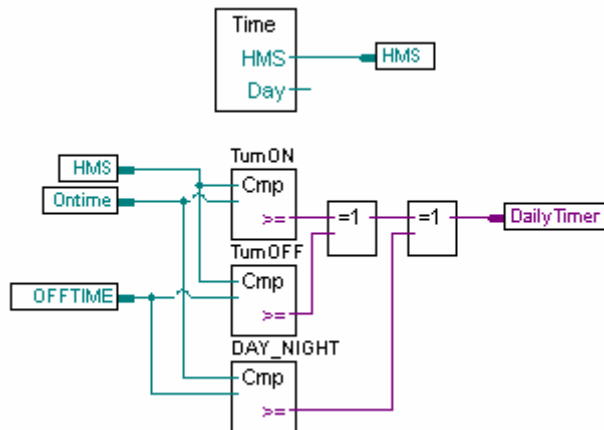
A questo punto possiamo iniziare la programmazione. All'inizio di questo capitolo avevamo creato un progetto contenente un file denominato: *DailyTimer.fup*. Questo è il file in cui ora scriveremo il presente esempio di programmazione.



Iniziamo creando l'elenco dei simboli. Notare che l'ora corrente del PCD viene salvata in un registro dinamico HMS. L'indirizzo di questo registro non è ancora stato definito. Il pacchetto PG5 provvederà automaticamente ad assegnare un proprio indirizzo nella fase di costruzione del programma.

La stessa cosa avviene per le ore di accensione e spegnimento (*ONTIME*, *OFFTIME*), ad eccezione del fatto che «:=60000» non è l'indirizzo di un registro, bensì il valore con cui questo verrà inizializzato durante il trasferimento (download) del programma nel PCD (:=60000 significa 6 ore 00 minuti 00 secondi).

N.B.: Una partenza a freddo del PCD non reinizializza questi registri. Essi possono essere reinizializzati esclusivamente eseguendo il download del programma!



Tutti gli FBox necessari possono essere reperiti nel gruppo *Standard* della finestra *Selettore FBox*:

- *Time related, Read time (Temporizzatori, Leggi ora)*
- *Integer, Is greater or equal to (Intero, maggiore o uguale a) Binary, Xor (Binario, Xor)*

### 4.12 Costruzione (Build) del programma



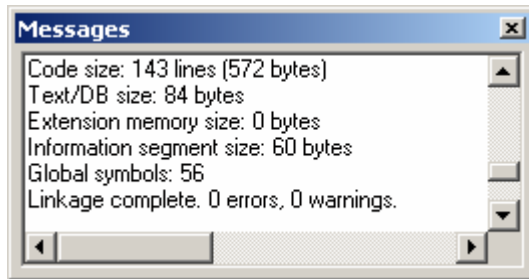
Costruisci  
Tutto

Prima che il programma appena completato possa essere letto ed eseguito dal PCD, è necessario “costruirlo” (o convertirlo) utilizzando il comando di menu CPU Build (CPU Costruisci) oppure il pulsante Build (Costruisci).

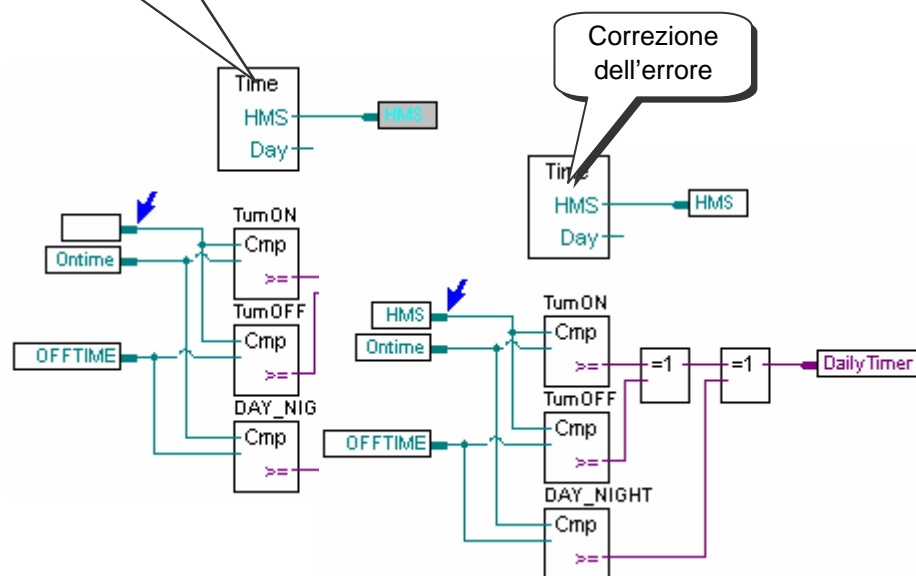
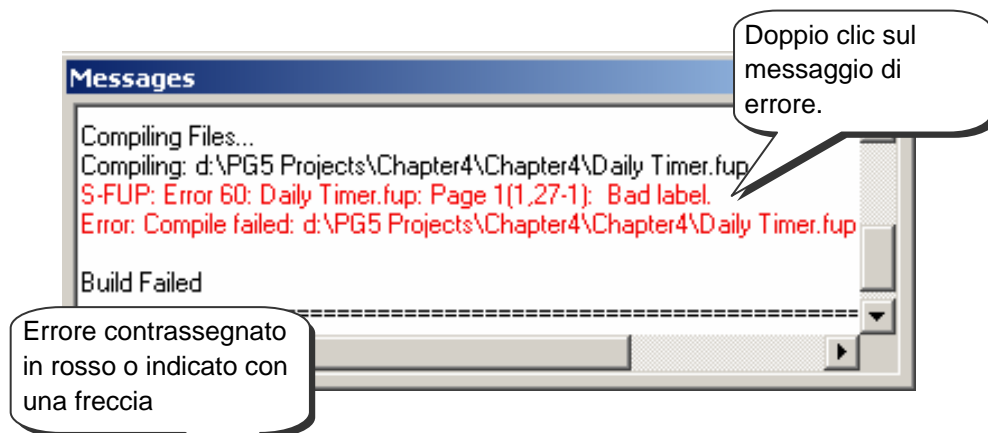
La finestra Messages visualizza i risultati dei vari stadi di costruzione del programma (Compilazione, Assemblaggio, Link, ecc.).

Se il programma è stato scritto correttamente, la funzione “build” si conclude con il messaggio: Build successful. Total errors 0 Total warnings: 0

(Costruzione eseguita con successo. Totale errori 0 Totale avvertimenti 0)



Gli eventuali errori generati in fase di costruzione vengono indicati con un messaggio in rosso. La localizzazione dell'errore nel programma utente può essere effettuata facilmente facendo doppio clic con il pulsante del mouse sul messaggio di errore.



### 4.13 Trasferimento (Download) del programma nel PCD



Trasferimento  
Programma

A questo punto il programma utente è pronto. Non resta che eseguire il trasferimento del programma stesso (download) dal PC al PCD. Questo può essere effettuato utilizzando il pulsante *Download Program (Trasferisci Programma)* oppure attraverso la finestra *SAIA Project Manager*, menu *Online, Download Program (In linea, Scaricamento Programma)*.

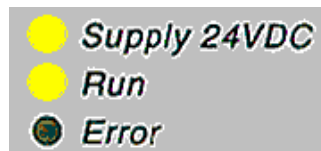
Nel caso in cui si verificano dei problemi di comunicazione, controllare i parametri di configurazione (*Settings Online (Impostazioni In Linea)* e *Settings Hardware (Impostazioni Hardware)*) e il cavo di collegamento (PCD8.K111, USB) tra PC e PCD.

### 4.14 Individuazione e correzione degli errori (Debug)

La prima versione di un programma non è sempre perfetta. È sempre necessario eseguire un test esaustivo. Il programma di test è supportato dallo stesso editore utilizzato per scrivere il programma.

#### 4.14.1 Pulsanti On/Offline – Run – Stop - Step-by-step

1. Premere il pulsante *Go On /Offline (In Linea/Fuori Linea)*
2. Avviare il programma con il pulsante *Run (Esegui)*



Contemporaneamente osservare il LED RUN sul PCD. Dopo aver premuto il pulsante Run, il Led RUN dovrebbe essere acceso. Questo significa che il PCD sta eseguendo il programma utente.

3. Quando si seleziona il pulsante *Stop*, il LED *RUN* si spegne e il PCD interrompe l'esecuzione del programma utente.
4. Il PCD esegue un FBox ad ogni pressione del pulsante *Step-by-step (Passo a Passo)* o del tasto *F11*.



Osservare l'indicazione *Stop* che indica l'avanzamento passo-passo del programma.

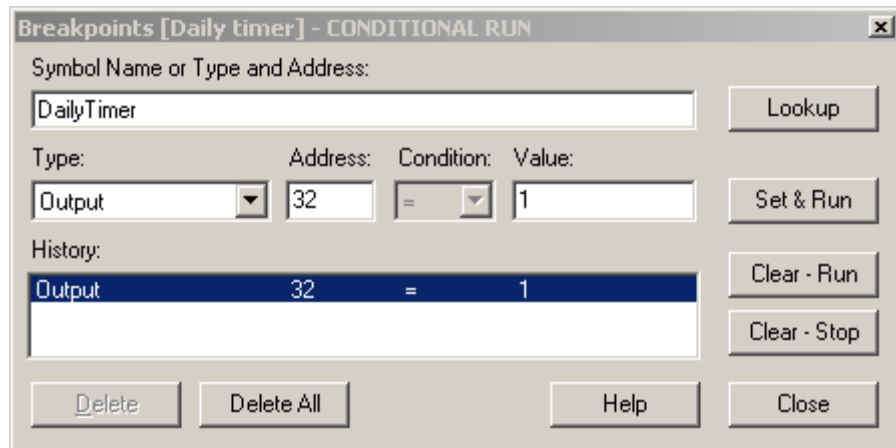
#### 4.14.2 Breakpoint (Interruzioni)

I Breakpoint (Interruzioni) permettono di arrestare l'esecuzione di un programma al verificarsi di un evento collegato ad uno degli FBox presenti nel programma stesso o a un simbolo:

- Stato alto o basso di un ingresso, uscita, flag o flag di stato
- Valore presente in registri o contatori

„Breakpoint” su simbolo

La condizione di arresto può esser definita mediante la voce di menu *Online Breakpoints (Breakpoint In Linea)*.



La finestra di dialogo sopra raffigurata permette di specificare il tipo di simbolo (Type) nonché l'indirizzo/numero associato (Address). E' anche possibile semplicemente trascinare il simbolo interessato dall'editore dei simboli al campo *Symbol Name (Nome Simbolo)*; a questo punto è sufficiente indicare la condizione per l'interruzione (Condition) ed il valore/stato (Value) desiderati.

Con il pulsante *Set & Run (Imposta e Passa in Run)* si forza il PCD in modalità RUN condizionato. Il LED *Run* del PCD lampeggerà ed il pulsante *Run* dedicato verrà visualizzato alternativamente in colore verde e rosso. Il PCD passerà automaticamente nello stato Stop quando la condizione di interruzione predefinita risulterà soddisfatta. Nell'esempio, quando un'istruzione modifica il valore dell'uscita 32, il suo stato passerà a livello alto (1), soddisfacendo la condizione di interruzione. L'ultimo Fbox elaborato dal PCD (che ha generato la condizione richiesta) verrà visualizzato in colore rosso. A questo punto è possibile continuare l'esecuzione del programma sia passo-passo che mediante un'altra condizione di interruzione (Breakpoint). Se necessario, è possibile interrompere il funzionamento in modalità RUN condizionato:

- Il pulsante *Clear-Run (Annulla-Run)* forza il PCD in modalità RUN. Il LED *Run* si illuminerà con luce fissa ed il pulsante di *Run* verrà visualizzato in colore verde
- Il pulsante *Clear-Stop (Annulla-Stop)* forza il PCD in modalità STOP. Il LED *Run* si spegnerà ed il pulsante di *Run* verrà visualizzato in colore rosso.

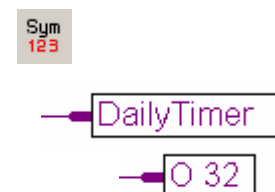
Se si sono definite più interruzioni (breakpoint) condizionali, esse verranno inserite nel campo *History (Storico)*. In questo caso, è possibile selezionare con il mouse una qualsiasi interruzione predefinita ed attivarla con il pulsante *Set & Run (Imposta e Passa in Run)*.

„Breakpoint” su FBox

Selezionare un FBox all'interno del programma quindi selezionare la voce di menu *Online, Run to, Fbox (In Linea, Esegui Fino A, FBox)* per impostare l'interruzione del programma al raggiungimento dell'FBox interessato. A questo punto, procedere con l'esecuzione del programma in modalità passo-passo.

#### 4.14.3 Visualizzazione dei simboli o degli indirizzi

Il pulsante *Show Operand as symbol or value (visualizza operando come simbolo o valore)* permette di visualizzare le informazioni presenti nei connettori, con i relativi simboli o indirizzi. Se, premendo il suddetto pulsante il simbolo non viene sostituito dall'indirizzo corrispondente, significa che l'indirizzo di tale simbolo viene assegnato dalla funzione build.



### 4.14.4 Visualizzazione dello stato dei simboli in Fupla

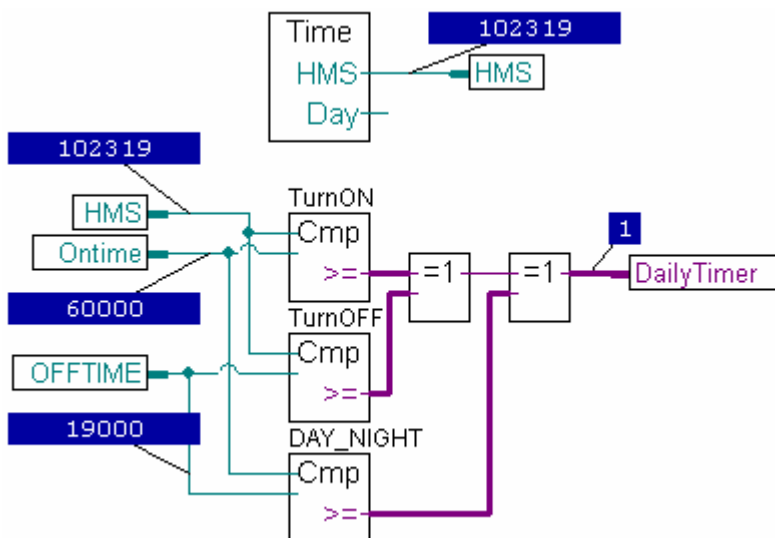
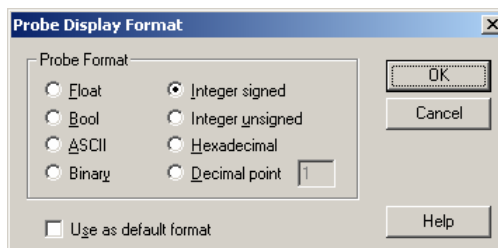
Quando l'editore è *In Linea (Online)* e il PCD è in modalità *RUN*, ogni singola risorsa utilizzata dal programma può essere visualizzata:



Posiziona Sonda

- Lo stato logico di una informazione binaria viene indicato con una linea spessa o sottile (spessa = 1 e sottile = 0)
- Tutte le altre informazioni possono essere visualizzate facendo clic con il pulsante sinistro del mouse sul collegamento desiderato.

Facendo doppio clic su una sonda, si apre la finestra *Probe Display Format (Formato di Visualizzazione Sonda)*, che permette di scegliere il formato in cui visualizzare i valori interessati: intero, esadecimale, binario, virgola mobile, booleano o ASCII.

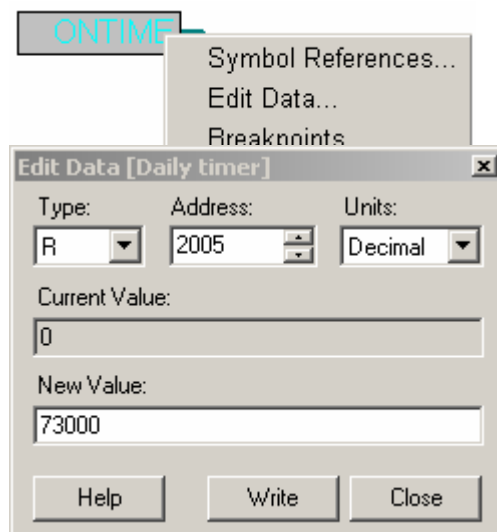


### 4.14.5 Modifica di simboli „In Linea“

Durante la verifica del comportamento del programma in determinate condizioni di utilizzo, può essere utile a volte modificare lo stato/valore dei simboli associati ai connettori di ingresso:

Selezionare il connettore di ingresso interessato quindi fare clic con il tasto destro del mouse per visualizzare il menu contestuale.

La voce di menu *Edit Data (Modifica Dati)* permette di modificare il valore/stato del simbolo associato al connettore interessato.





### 4.14.6 Visualizzazione dello stato dei simboli con la “Finestra di Monitoraggio” (Watch Window)

Un altro utile modo per verificare e visualizzare gli stati dei simboli nel nostro esempio, è quello di utilizzare la funzione *Watch Window (Finestra di Monitoraggio)*. Premere il pulsante di attivazione dedicato quindi trascinare i simboli dall’editore dei simboli alla Finestra di Monitoraggio.

1. Posizionare il mouse al centro dell'icona dei simboli. Premere il tasto sinistro del mouse.

2. Tenere premuto il tasto del mouse e trascinare il simbolo nella “Watch window” (Finestra di Monitoraggio).

3. Elenco dei simboli con commenti e stati.

4. Avvio/Arresto Monitoraggio

Symbol	Address	Value	Comment
HMS	R 1	233250	PCD Clock with current time
DailyTimer	O 32	0	Daily Timer
ONTIME	R 3	60000	Switch on time
OFFTIME	R 2	19000	Switch off time

Per controllare il corretto funzionamento del nostro esempio relativo al temporizzatore giornaliero, proveremo ora a modificare le istruzioni di accensione/spegnimento (*ONTIME* e *OFFTIME*), osservando lo stato dell’uscita *DailyTimer*. Per modificare l’istruzione, procedere come segue:

1. Avvio/Arresto Monitoraggio

2. Posizionare il cursore del mouse sul valore da modificare. Fare doppio clic con il tasto sinistro del mouse ed inserire il nuovo valore.

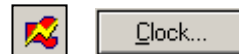
3. Trasferire i valori

Symbol	Address	Value	Modify Value	Symbol Comment
HMS	R 2113	104852		PCD Clock with current time
DailyTimer	O 32	1		Daily Timer
Ontime	R 2115	60000	43000	Switch on time
OFFTIME	R 2114	19000		Switch off time

### 4.14.7 Impostazione dell’orologio PCD

Quando si mette in servizio il PLC, il suo orologio interno non è sempre regolato sull’ora corretta. Per regolarlo, operare come segue:

1. Selezionare il pulsante *Online Configurator (Configuratore Online)* nella finestra *Project Manager*, quindi selezionare *Clock (Orologio)*.
2. Copiare l’ora dal PC al PLC con il pulsante *Copy to PCD>>> (Copia su PCD>>>)*, oppure regolare le impostazioni dell’orologio nei campi *SAIA PCD Clock (Orologio SAIA PCD)*.



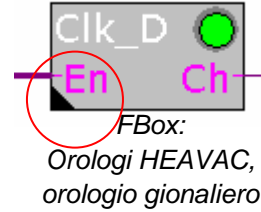
Set SAIA PCD Clock

PC Clock  
Date: 10/01/03  
Time: 11:23:51  
[Copy to PCD >>>]

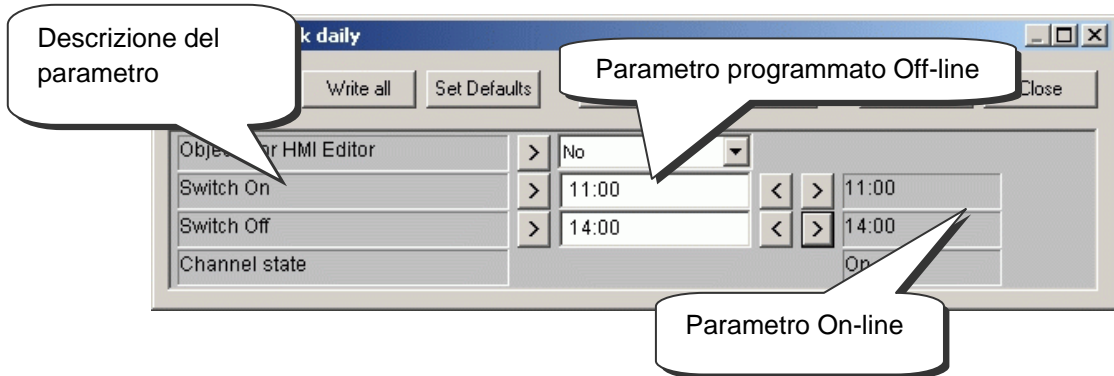
SAIA PCD Clock  
Date (dd/mm/yy): 10/01/2003  
Time (hh:mm:ss): 0:28:46  
Day of week (1-7): 5  
Week of year (1-53): 2  
[OK] [Cancel]

### 4.15 Finestra di Configurazione

Alcuni FBox, identificati da un triangolo nero nell'angolo in basso a sinistra, supportano una finestra di dialogo con un certo numero di parametri configurabili. Questi parametri configurabili sono frequentemente utilizzati nelle librerie HEAVAC, ed anche in altre librerie di FBox.



I parametri inclusi nelle finestre di configurazione definiscono le variabili nello stesso modo dei collegamenti in ingresso, ma con benefici aggiuntivi quali: descrizione dei singoli parametri, possibilità di correzione on-line, semplicità d'uso, ecc. Per visualizzare la finestra di configurazione, fare doppio clic con il pulsante sinistro del mouse su uno qualsiasi degli FBox contenente il triangolo nero.



La finestra di configurazione ha tre colonne:

**Colonna con la descrizione del parametro configurazione.** Descrive l'utilizzo del parametro. Se la descrizione termina con tre puntini, significa che sono disponibili informazioni aggiuntive - Facendo doppio clic sul testo con il pulsante sinistro del mouse.

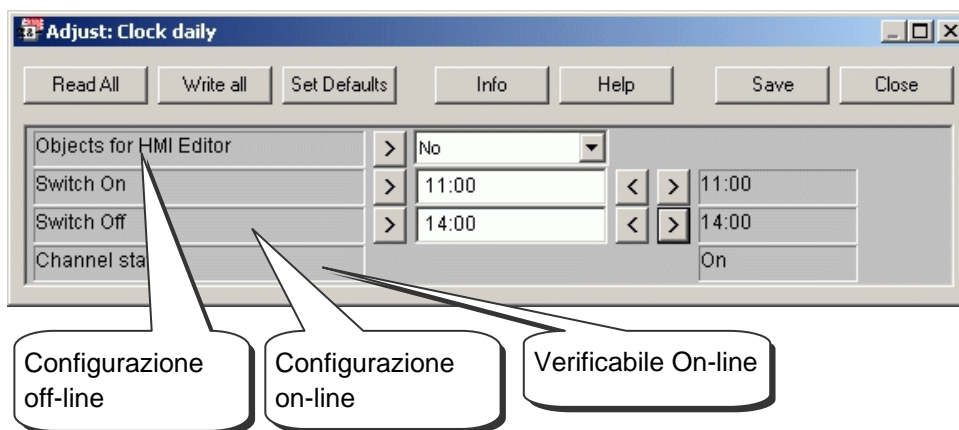


**Colonna con il valore del parametro Off-line.** Questa è il valore che era stato definito in fase di programmazione dell'applicazione e che è stato salvato nel file Fupla. Il parametro può essere generalmente definito sotto forma di intero, selezione a scelta multipla o, a volte, anche sotto forma di pulsante. Per alcuni parametri, il valore può essere modificato anche quando il PCD è in modalità on-line.



**Colonna con il valore On-line.** Questa colonna visualizza le informazioni salvate nella memoria del PCD (registro o flag). Rappresenta i dati utilizzati dal programma PCD quando quest'ultimo si trova in modalità Run.

### 4.15.1 Tipi di Parametri Configurazione



I parametri configurabili possono essere suddivisi in tre gruppi principali.

#### Parametri configurabili Off-line

Ogni modifica off-line di un parametro richiede l'esecuzione di un comando *Build All (Costruisci Tutto)* o *Download Program (Trasferimento Programma)* per far sì che il programma PCD lo possa prendere in considerazione.

#### Parametri configurabili On-line

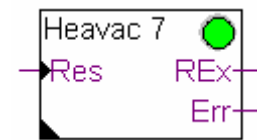
La modifica on-line di un parametro non richiede l'esecuzione di un comando *Build All (Costruisci Tutto)* o *Download Program (Trasferimento Programma)* per poter essere preso in considerazione dal programma PCD. Il parametro viene infatti modificato direttamente nella memoria del PCD.

#### Parametri verificabili On-line

Questi parametri non prevedono la possibilità di modifiche. Possono essere visualizzati a solo scopo informativo e per verificare il corretto funzionamento di un programma o FBox.

### 4.15.2 Inizializzazione degli FBox HEAVAC

Quando si utilizzano determinate librerie di FBox, ad esempio le applicazioni HEAVAC, è necessario posizionare sempre un FBox di inizializzazione all'inizio del file Fupla. Questo permette di gestire alcune delle attività comuni della libreria, ad esempio l'inizializzazione della libreria stessa dopo il trasferimento del programma o dopo una ripartenza a freddo del PCD (PCD power-up).



*Fbox:  
Inizializzazione HVC,  
Inizializzazione HVC 7*

Dopo una qualsiasi operazione di trasferimento del programma e ripartenza a freddo del PCD, l'ingresso *Res* di questo FBox ed i parametri di configurazione illustrati nella figura seguente esercitano una importante influenza sull'inizializzazione dei parametri di configurazione per tutti gli altri FBox HEAVAC utilizzati nel programma.



#### **Trasferimento del programma e parametro Automatic Reset (Reset Automatico):**

Con l'opzione Attivata, i parametri di configurazione di tutti gli FBox HEAVAC verranno inizializzati con i valori definiti dal programma.

Con l'opzione Non attivata, tutti i parametri esistenti nel PCD verranno mantenuti.

#### **Ingresso Res e parametro Evaluate Reset Input (Valutazione Ingresso di Reset):**

Se lo stato dell'ingresso di reset è alto, i parametri di configurazione di tutti gli FBox HEAVAC verranno inizializzati con i valori definiti in fase di programmazione.

In base all'opzione selezionata per il parametro Evaluate Reset input (Valutazione Ingresso di reset), l'ingresso Res verrà preso in considerazione solo in caso di ripartenza a freddo del PCD oppure durante l'esecuzione del programma (cioè sempre).

#### **LED Verde/rosso**

Alcuni FBox hanno un LED simulato che può assumere tre colori diversi: grigio quando il controllore è off-line, verde o rosso quando il controllore è on-line. Verde significa che tutto sta funzionando correttamente, mentre rosso indica un errore (generalmente causato da informazioni presenti sugli ingressi dell'FBox oppure dalla selezione di parametri di configurazione non appropriati. Per informazioni più dettagliate, si prega di consultare le guide riguardanti gli errori Fbox).

#### **N.B.:**

All'interno della libreria HEAVAC esistono differenti versioni della funzione di inizializzazione (Inizializzazione HEAVAC 4, ...7). La versione 7 è la più recente. Per tutte le nuove applicazioni si raccomanda di utilizzare la versione 6.

### 4.15.3 FBox HEAVAC con parametri di configurazione

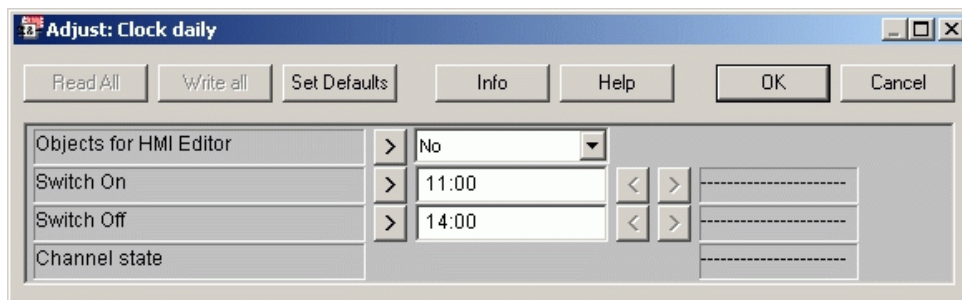
L'FBox *CIK\_D* permette di creare un orologio giornaliero simile a quello creato all'inizio di questo capitolo, ma con un singolo FBox disponibile nella libreria HEAVAC.

L'uscita dell'FBox può essere attivata (on) o disattivata (off) in base ai tempi definiti nella finestra di regolazione.

Il parametro *Objet for HMI Editor (Oggetti per editore HMI)* deve essere utilizzato solo in presenza di terminali HMI. Se questa opzione non viene utilizzata, mantenere il parametro standard proposto. L'ingresso *En* permette di disabilitare la funzione orologio. Se *En* è basso, l'uscita *Ch* rimarrà inattiva.



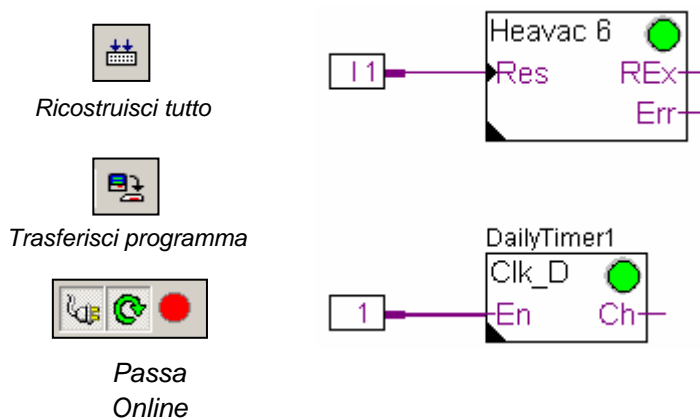
FBox: orologi HEAVAC, Orologio giornaliero



### 4.15.4 Applicazione Mini HEAVAC

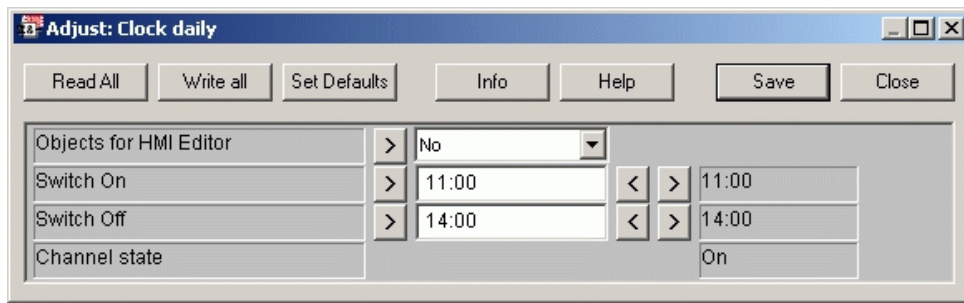
Per verificare il funzionamento dei parametri previsti nella finestra di configurazione, possiamo utilizzare ancora una volta il programma dell'orologio giornaliero presentato all'inizio di questo capitolo. Questa volta, tuttavia, lo realizzeremo con l'ausilio della Libreria HEAVAC.

I due FBox descritti in precedenza sono gli unici Fbox necessari. Creare il programma come indicato qui di seguito, quindi eseguire *ReBuild All, Download Program and Go Online (Ricostruisci tutto, Trasferisci Programma, e Passa in stato Online)*.



Se il programma viene esteso con l'utilizzo di diversi altri FBox, l'Fbox *Inizializzazione HEAVAC 7*, deve essere posizionato una sola volta all'inizio della prima pagina Fupla.

### 4.15.5 Parametri dopo il trasferimento del programma

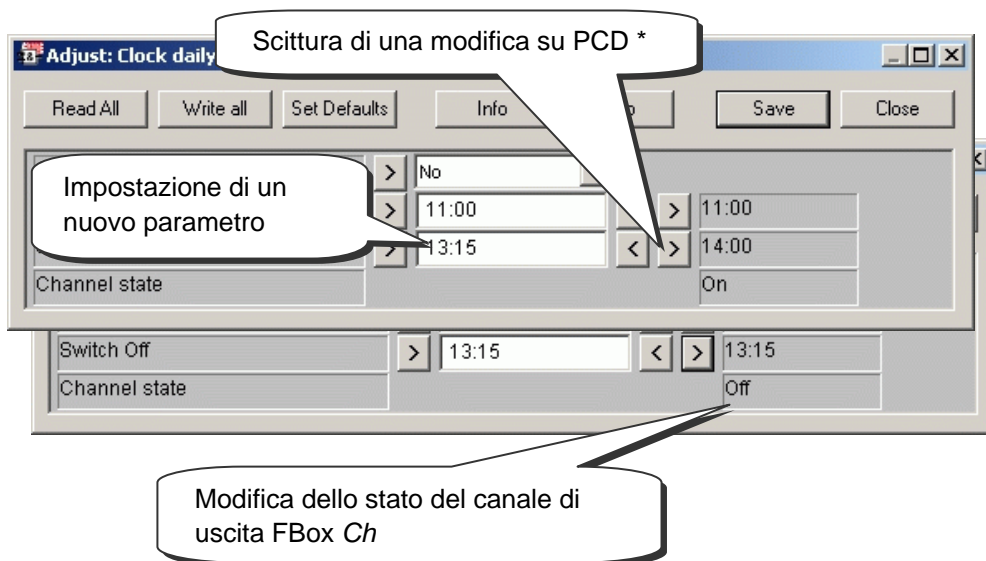


La colonna centrale indica i parametri di attivazione (switch-on) e disattivazione (switch-off) dell'orologio giornaliero, definiti durante la fase di programmazione.

Come regola generale<sup>1</sup>, questi parametri dovranno coincidere con quelli utilizzati nel PCD e visualizzati nella colonna di destra. (parametri switch-on/switch-off e stato dell'uscita *Ch*)

### 4.15.6 Scrittura dei parametri on-line

Durante il collaudo on-line, è possibile impostare nuovi valori nei parametri "switch-on" e "switch-off" per il canale di uscita dell'FBox:



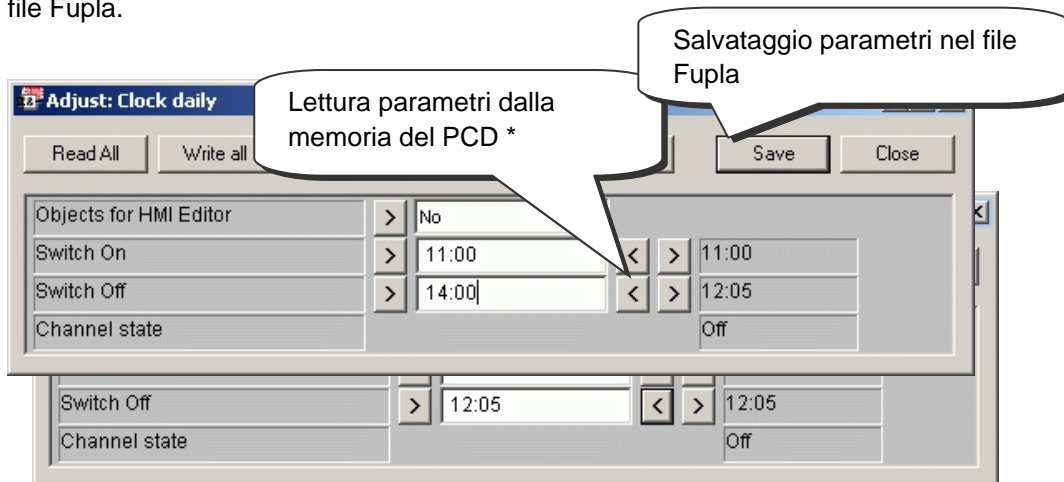
\* E' possibile scrivere tutti i parametri della finestra di configurazione anche selezionando il pulsante *Write All (Scrivi Tutto)*.

Se i parametri corretti devono essere salvati nel programma Fupla per il prossimo comando *Build All (Costruisci Tutto)*, chiudere la finestra di configurazione con il pulsante *Save (Salva)*, altrimenti usare semplicemente il pulsante *Close (Chiudi)*.

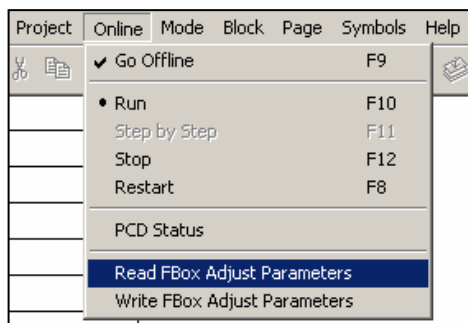
<sup>1</sup> Questa regola può variare in base alle opzioni definite nella finestra di configurazione *Heavac 7* ed allo stato dell'ingresso *Res*, durante la ripartenza a freddo del PCD.

### 4.15.7 Lettura dei parametri on-line

A volte l'utente potrebbe voler leggere e salvare i parametri di configurazione esistenti nella memoria del controllore, per il prossimo comando *Build All (Costruisci Tutto)*. Questa operazione provvederà a trasferire i parametri della memoria del controllore al file Fupla.

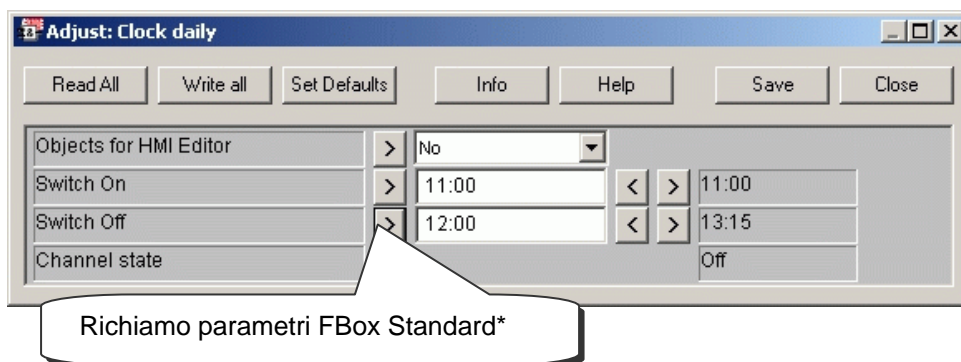


- \* E' possibile leggere tutti i parametri della finestra di configurazione anche premendo il pulsante *Read All (Letti Tutto)*.
- \* Inoltre è possibile leggere tutti i parametri di configurazione di tutti gli FBox inclusi nel file Fupla caricato nella memoria del PCD, selezionando il comando *Read Fbox, Adjust parameter (leggi Parametri di configurazione Fbox)* nel menu *Online*.



### 4.15.8 Ripristino dei parametri di default

Anche dopo numerose modifiche apportate ai parametri, esiste sempre la possibilità di ripristinare i parametri di default. Questi sono i parametri definiti quando l'Fbox era stato inserito per la prima volta nella pagina Fupla.



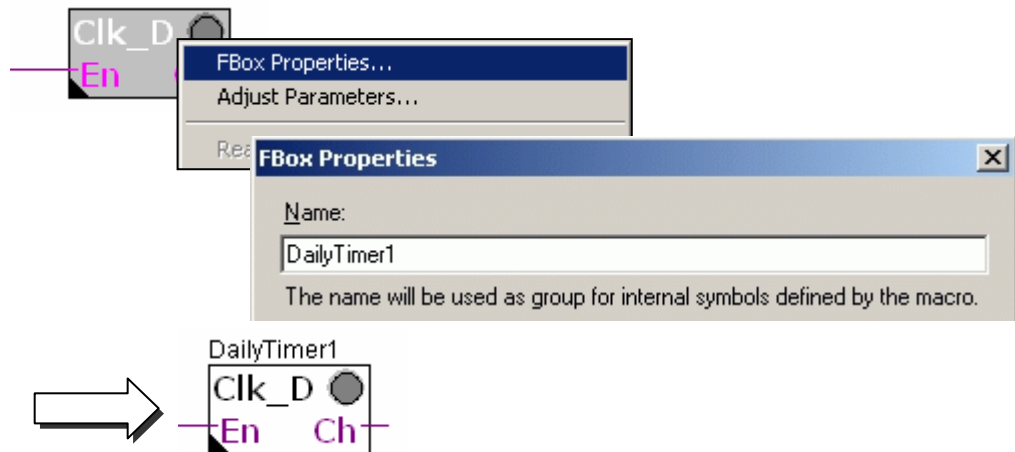
- \* E' possibile ripristinare tutti i parametri di default nella finestra di configurazione, anche selezionando il pulsante *Set Defaults (Imposta Default)*.

### 4.15.9 Definizione di simboli per i parametri di configurazione

A volte può essere necessario leggere o scrivere dei parametri nella finestra di configurazione dal programma Fupla, da una rete di comunicazione, o dal sistema di supervisione.

Questo è possibile se sono stati definiti dei simboli per Flag o registri corrispondenti ai parametri visualizzati nella finestra di configurazione FBox.

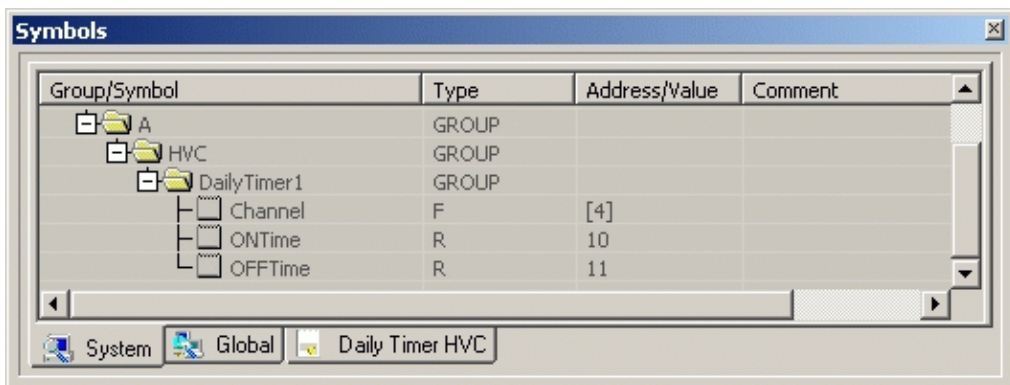
Per definire questi simboli, fare clic con il pulsante destro del mouse sull'FBox in modo da attivare il menu contestuale. Selezionare la voce *FBox Properties...* (*Proprietà Fbox*). Definire un nome simbolico per un gruppo di parametri linkati all'Fbox selezionato.



  
Ricostruisci tutto

Costruire il programma ed aprire l'editore dei simboli. A questo punto comparirà una nuova directory *System* contenente l'elenco dei simboli di sistema del PCD.

Nella libreria HEAVAC, tutti i simboli di sistema corrispondenti ai parametri della finestra di configurazione sono raggruppati sotto *A.HVC.name* (dove *name* è il nome dell'FBox).



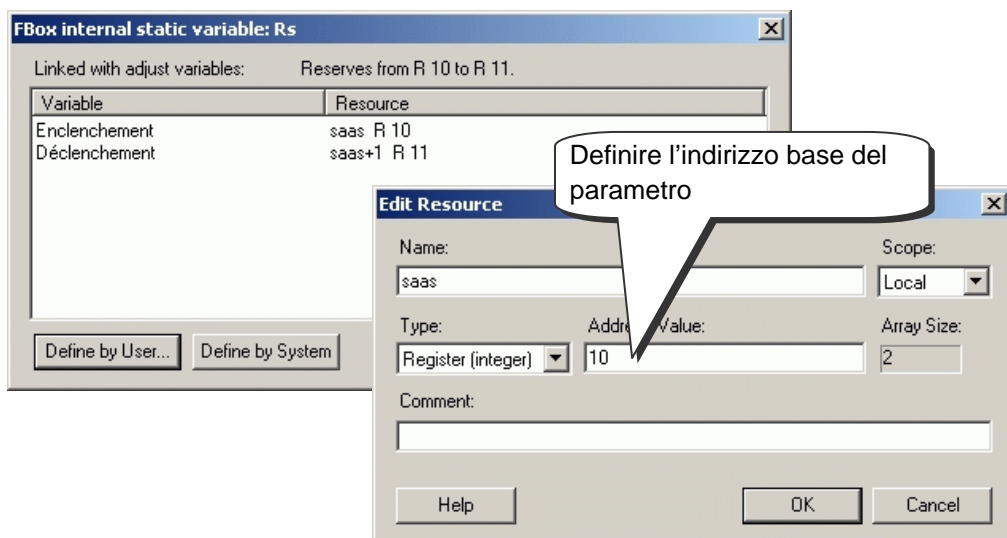
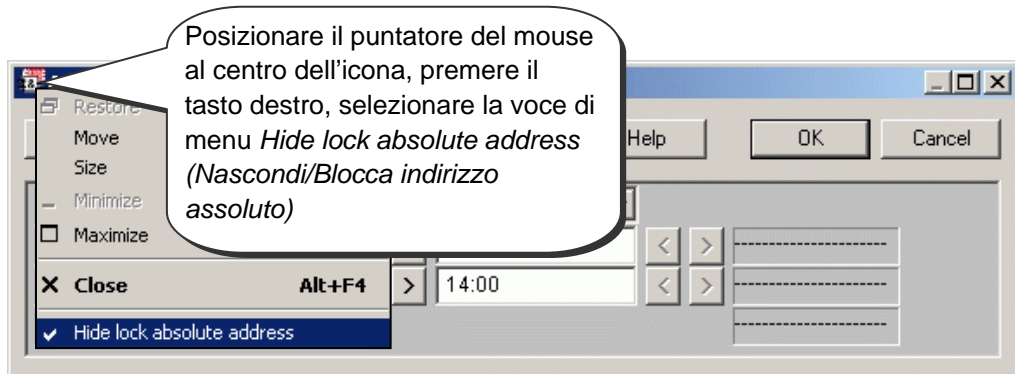
Questi nuovi simboli potranno ora essere liberamente utilizzati nel programma Fupla.





### 4.15.10 Definizione degli indirizzi dei parametri di configurazione

Definire il simbolo del parametro di configurazione come descritto in precedenza ed aggiungere l'indirizzo operando come segue:



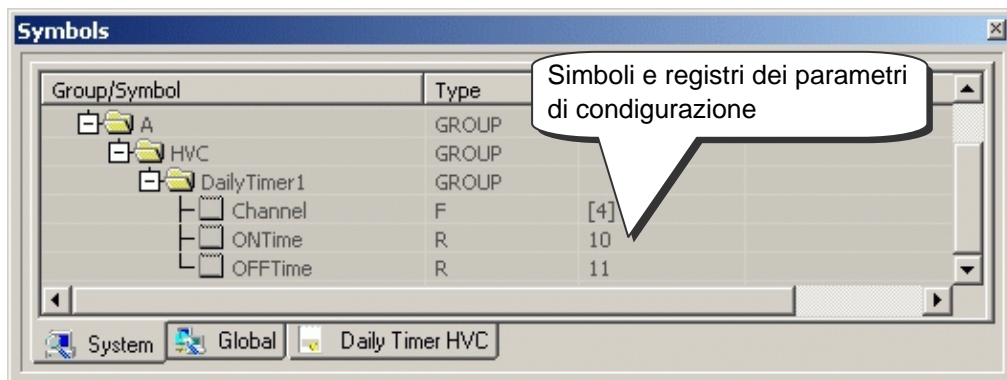
Costruire il programma ed aprire l'editore dei simboli. Ai simboli di sistema sono stati assegnati gli indirizzi dei registri illustrato nella figura seguente.



Ricostruisci Tutto



Mostra/  
Nascondi  
Editore dei  
Simboli



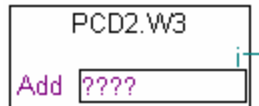
## 4.16 Messa in servizio di un modulo analogico

### 4.16.1 Acquisizione di una misura analogica

I programmi di esempio presentati fino ad ora hanno utilizzato ingressi ed uscite digitali, con l'inserimento degli indirizzi o dei simboli nel margine dell'editore FUPLA.



Con i moduli di ingresso o di uscite analogiche, per l'acquisizione del valore analogico deve essere utilizzato un FBox. Questi FBox sono disponibili con le librerie: *Standard, Moduli Analogici, Applicazioni, e Analogiche-HEAVAC*.



Queste librerie offrono un' ampia varietà di FBox, ognuno corrispondente ad uno specifico modulo analogico. Il nome che appare nel *selettore degli FBox* definisce il numero di articolo del modulo.

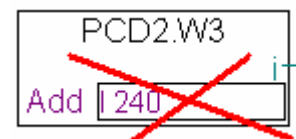
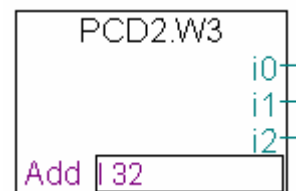
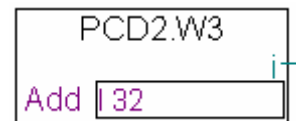
Gli Fbox Analogici sono espandibili. L'utente può definire il numero dei canali di misura richiesti da ogni applicazione. Se alcuni canali di misura non sono utilizzati, oppure viene inserito un nuovo canale, per adattare la dimensione può essere utilizzato il menu contestuale *Resize FBox*. Un FBox, comunque, può essere definito con il massimo numero di canali, anche se non tutti verranno utilizzati.

Il campo *Add* permette di definire l'indirizzo di base del modulo analogico. Questo indirizzo indica dove il modulo è stato inserito nel PCD : 0, 16, 32, ...

Le misure analogiche sono disponibili sugli ingressi da i 0 a i 7 del FBox. Questi possono essere connessi direttamente agli altri FBox, oppure il valore può essere salvato in un registro. Salvare il valore in un registro è una buona soluzione, particolarmente quando il valore sarà utilizzato in diverse pagine del programma.

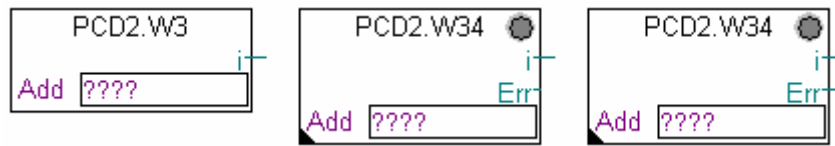
#### Attenzione:

Fare attenzione di non definire mai più di un FBox per ogni modulo analogico, e di non inserire mai il modulo analogico all'indirizzo del watchdog del PCD (255). Altrimenti, il valore fornito dal modulo potrebbe essere non corretto.



### 4.16.2 Esempio per i moduli di ingressi analogici PCD2.W340

Se il PCD è equipaggiato con un modulo PCD2.W340, che è fornito di 8 canali di ingresso universali, l'utente può utilizzare uno dei seguenti Fbox FUPLA e definire il numero di canali di misura richiesti.



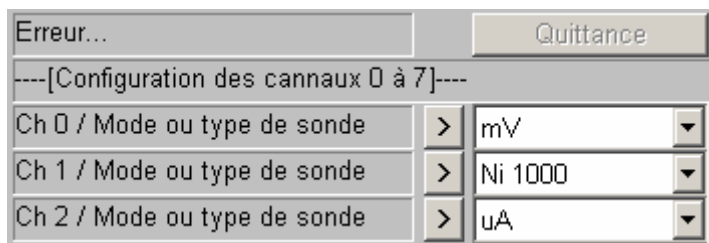
FBox: PCD2.W3, PCD2.W34, PCD2.W34 con errore

Le unità di misura dipendono dal modulo, dal FBox, e dai parametri selezionati.

Il PCD2.W340 è un modulo universale. Supporta misure nei campi 0..10V, 0..2.5V, 0..20 mA e sensori di temperatura Pt/Ni 1000. Un ponticello sul modulo permette di selezionare il campo di misura. La risoluzione è di 12 bit, equivalenti a 4095 distinti stati di misura. (Per informazioni dettagliate riguardanti questi moduli, fare riferimento al manuale hardware del vostro PCD).

L FBox *PCD2.W3* fornisce una misura non ingegnerizzata. Questo modulo ha la risoluzione di 12 bit, corrispondente ad un valore di misura tra 0 e 4095. L'utente ha così il compito di convertire la misura in una unità fisica standard.

L Fbox *PCD2.W34* è più elaborato. E' provvisto di una finestra di personalizzazione che permette di definire l'unità di misura per ogni canale. Il LED del FBox diventa rosso se almeno una delle misure supera la validità del campo: corto-circuito o rottura nel cavo del sensore. L'errore può essere riconosciuto con il pulsante *Acknowledge* nella finestra di personalizzazione.



L'FBox *PCD2.W34 con errore* offre gli stessi servizi per la conversione di unità, inoltre ha un'uscita d'errore che indica quale canale è in errore, in più ha un ulteriore parametro di personalizzazione per definire un valore di default in caso di errore.



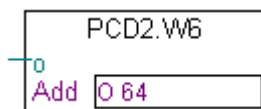
### 4.16.3 Esempio per i moduli di uscite analogiche PCD2.W610

Viene applicato lo stesso principio utilizzato per gli ingressi: l'utente inserisce nella pagina FUPLA un FBox che corrisponde al modulo di uscita analogico, lo trascina per selezionare il numero di canali di uscita e definire l'indirizzo di base del modulo.

A differenza degli Fbox di ingresso, i valori delle uscite analogiche sono visualizzate sul lato sinistro del FBox.

Questi ingressi possono essere collegati direttamente agli altri FBox, oppure ai registri definiti nel margine sinistro della pagina FUPLA.

Se il PCD è equipaggiato con un modulo PCD2.W610, che è fornito di 4 canali analogici universali, è possibile utilizzare l'FBox indicato sotto con uscite in corrente 0...20 mA, o in tensione 0...10 V.



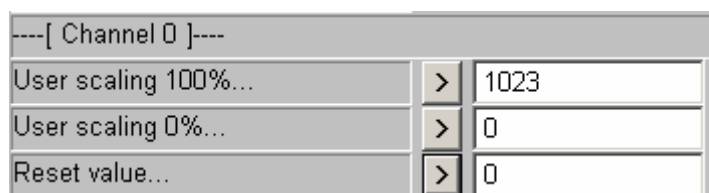
Fbox: PCD2.W6

Un ponticello sul modulo permette di selezionare il campo delle uscite. La risoluzione di questo modulo è di 12 bit, che corrispondono a 4095 distinti valori di stato.

Il valore intero all'ingresso dell'FBox determina la tensione o la corrente di uscita del canale:

Valore d'ingresso al Fbox	Tensione d'uscita [V]	Corrente d'uscita [mA]
0	0	0
2047	5	10
4095	10	20

Altri FBox hanno una finestra di personalizzazione per poter adattare il campo del valore di uscita applicato all'ingresso dell'FBox (es. FBox per il modulo PCD2.W605, che è fornito di 6 uscite elettricamente isolate 0...10 V):



I parametri *User scaling 0 and 100%* permettono di definire i valori minimi e massimi della tensione applicata al canale connesso all'ingresso dell'FBox.

Il parametro *Reset value* corrisponde al valore applicato al canale quando il PCD viene acceso.

## Indice

---

5	Strutture dei Programmi	5-3
5.1	Introduzione	5-3
5.2	Blocchi ad organizzazione ciclica (da COB 0 a 15)	5-4
5.2.1	Definizione	5-4
5.2.2	Esempio	5-5
5.2.3	Aggiunta di una struttura	5-5
5.2.4	Tempo di supervisione	5-6
5.3	Blocchi di Programma (da PB 0 a 299)	5-7
5.3.1	Definizione	5-7
5.3.2	Esempio	5-7
5.4	Blocchi Funzione (da FB 0 a 999)	5-9
5.4.1	Definizione	5-9
5.4.2	Esempio di chiamata funzione	5-9
5.5	Visualizzazione della Struttura e Percorso di Esecuzione	5-10
5.6	Blocchi ad Organizzazione Esclusiva (XOB)	5-11
5.6.1	Definizione	5-11
5.6.2	Elenco di tutti i blocchi XOB della famiglia PCD	5-12
5.6.3	Utilizzo dei Blocchi XOB	5-13
5.6.4	Tabella Storica	5-16
5.6.5	Descrizione degli XOB	5-17
5.7	Blocchi Sequenziali (da SB 0 a 31, 96)	5-20
5.8	Tabella riassuntiva.	5-21



## 5 Strutture dei Programmi

---

### 5.1 Introduzione

Il successo di un programma risiede nella sua struttura. Questa semplifica il programma stesso e ne velocizza lo sviluppo e la manutenzione. Il linguaggio di programmazione dei SAIA®PCD è un linguaggio strutturato che utilizza vari blocchi organizzativi per contenere le istruzioni richieste dall'applicazione. Ogni tipo di blocco fornisce diversi servizi all'utente. I blocchi organizzativi disponibili sono: blocchi ad organizzazione ciclica (COB), blocchi funzione (FB), blocchi programma (PB), blocchi ad organizzazione eccezionale (XOB) e blocchi sequenziali (SB).

## 5.2 Blocchi ad organizzazione ciclica (da COB 0 a 15)

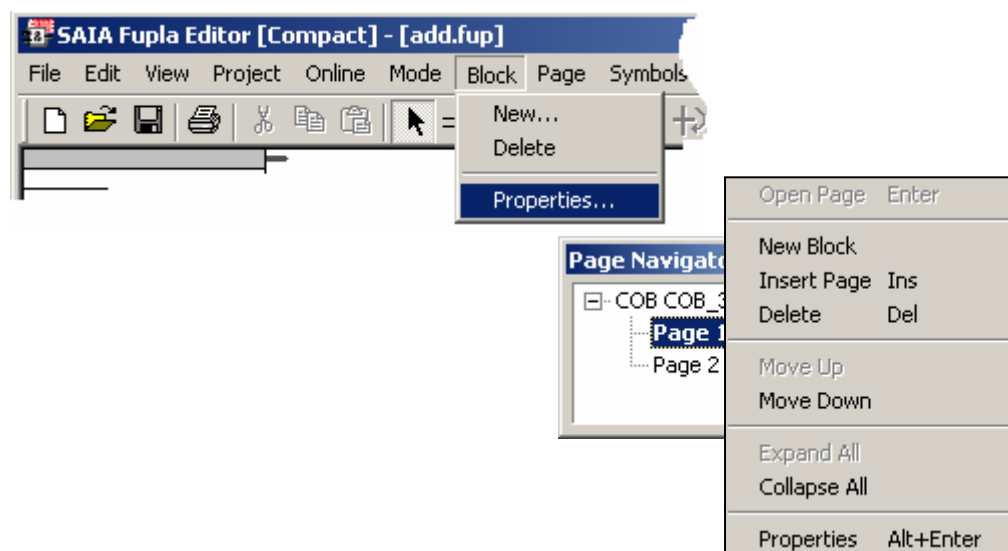
### 5.2.1 Definizione

I Blocchi ad Organizzazione Ciclica (COB) sono le parti del programma che vengono eseguite senza necessità di avere dei loop di programma e senza dover attendere eventi interni o esterni al PCD. All'avvio del PCD il programma esegue per prima cosa il COB 0. I COB da 1 a 15 vengono quindi eseguiti consecutivamente, se presenti nel programma. Essi vengono automaticamente richiamati in successione, in un loop continuo.

Tutti i segnali che devono essere trattati in maniera regolare (ad esempio i fine corsa per i movimenti dei motori, i segnali di interruzione dell'alimentazione esterna o i segnali per l'arresto di emergenza, i dispositivi per la protezione degli operatori ecc.) devono essere inseriti all'interno di un COB.

Nel PCD deve essere previsto almeno un COB!

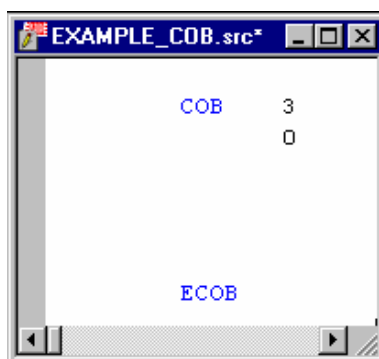
E' importante comprendere correttamente il concetto di blocco ad organizzazione ciclica. La mancanza di loop di attesa non è una limitazione, bensì una misura di sicurezza. Infatti è l'unico modo per garantire che i segnali importanti per l'applicazione vengano controllati ad intervalli regolari.



Se si scrive il programma utilizzando l'editore Fupla, i nuovi file vengono automaticamente aperti con un COB.

Utilizzando il menu *Block*, *Properties* (*Blocco*, *Proprietà*) è possibile modificare il tipo o il commento del blocco.

Nei programmi in Lista Istruzioni, (IL) il blocco è definito dalle Istruzioni che delimitano il programma.





### 5.2.2 Esempio

Ecco un esempio di programma (in IL e in FUPLA) che fa lampeggiare l'uscita 64 ogni 1,5 secondi. Il programma è scritto nel COB 0, che è seguito da altri COB, da 1 a 15.

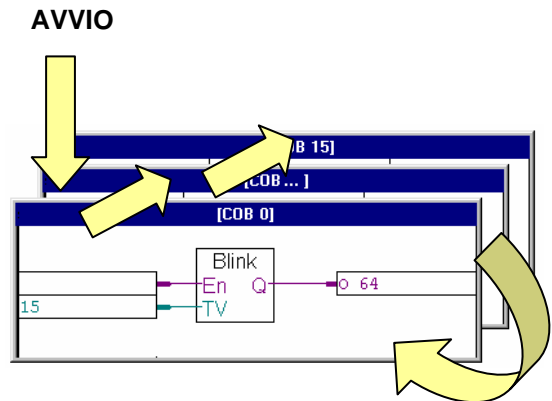
#### Programma in Lista Istruzioni:

```
COB 0 ; Avvio COB 0
      ; tempo di supervisione
STL T 1 ; Se temporizzatore 1 = 0,
LD T 1 ; carica con 1.5 s.
      15
COM O 64 ; e commuta l'uscita 64
ECOB ; Fine COB 0
```

```
COB 15 ; Prossimo blocco
      0
NOP
ECOB
```

; Prossimi COB

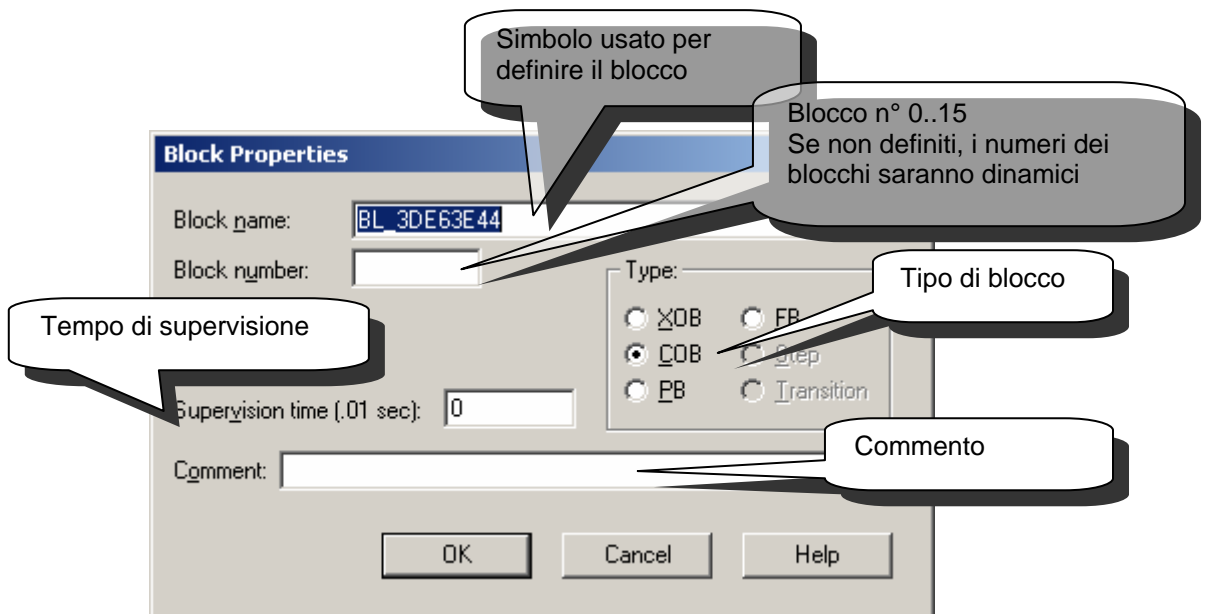
#### Programma in FUPLA:



Fbox: *Blinker, Blink delay T*  
(Lampeg., Ritardo di lampeggio T)

### 5.2.3 Aggiunta di una struttura

Un file Fupla può contenere diversi blocchi programma che possono essere aggiunti, cancellati o modificati utilizzando il menu *Block (Blocco)*.



#### 5.2.4 Tempo di supervisione

Il tempo di supervisione permette di definire il tempo massimo per l'elaborazione di un COB dall'inizio alla fine. Trascorso questo tempo, sono possibili due scenari:

Se non è stato programmato XOB 11, verrà abbandonato il COB attuale per elaborare i COB successivi, fino all'ultimo. La lampada di errore sarà accesa. Nel ciclo di programma successivo, il COB per il quale era scaduto il tempo di supervisione verrà riavviato con un nuovo tempo di supervisione, a partire dal punto in cui era stato interrotto.

Se è stato programmato XOB 11, verrà abbandonato il COB attuale per elaborare l'XOB 11. Al termine di quest'ultimo, il COB per il quale era scaduto il tempo di supervisione verrà riavviato con un nuovo tempo di supervisione, a partire dal punto in cui era stato interrotto. La lampada di errore non verrà accesa, in quanto l'errore era previsto e gestito dal programma utente.

Un tempo di supervisione uguale a zero, significa che il tempo di supervisione stesso è stato disattivato.

### 5.3 Blocchi di Programma (da PB 0 a 299)

#### 5.3.1 Definizione

È possibile anche lavorare con i PB (Blocchi di Programma). I PB permettono di organizzare il programma in modo gerarchico. I PB vengono attivati solo se vengono richiamati da un COB, PB, FB o SB (Blocco Sequenziale).

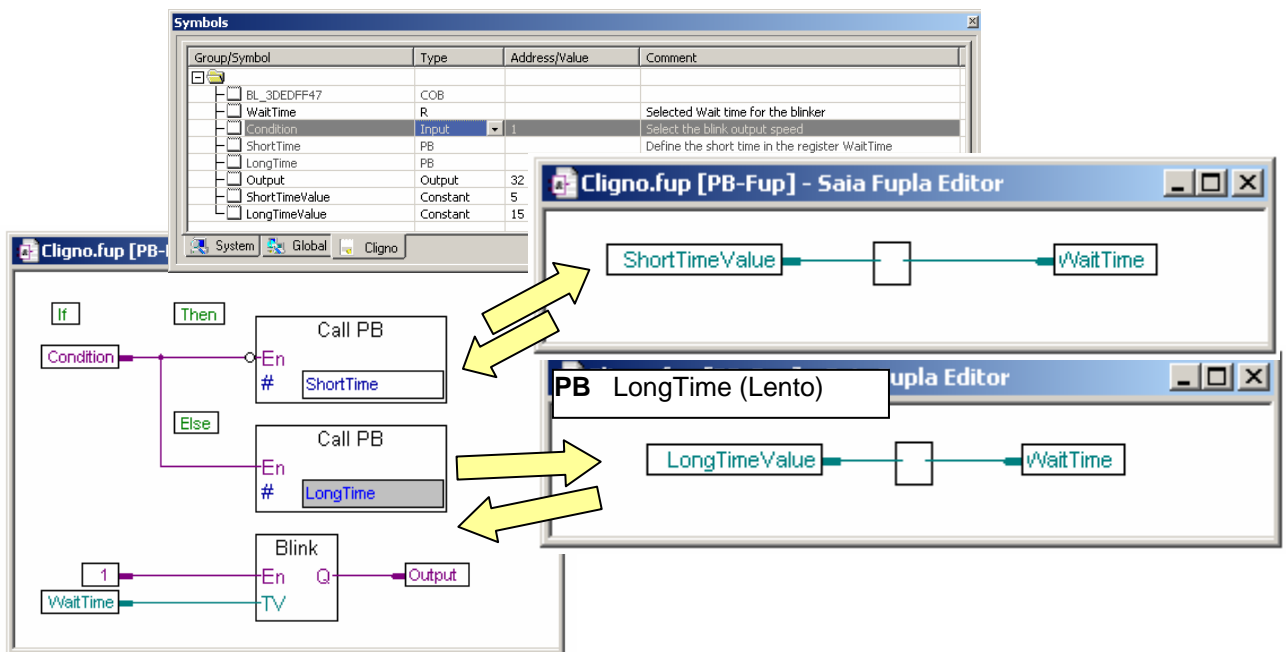
Un PB può essere richiamato in due modi: Condizionato o Incondizionato. Le chiamate condizionate dipendono dal risultato di una operazione logica. Nel programma, lo stesso PB può essere richiamato più volte. Un PB può richiamare un altro PB e così via fino a sette livelli di nidificazione.

Dopo il settimo livello di nidificazione, il PCD effettuerà una chiamata al blocco di gestione errori XOB 10.

#### 5.3.2 Esempio

Realizzazione di un lampeggiatore a doppia velocità in base allo stato logico dell'ingresso "Condition".

##### Esempio Fupla:



Se lo stato logico dell'ingresso digitale *Condition* è basso, il PCD chiamerà il blocco PB *ShortTime* e trasferirà la costante *ShortTimeValue* (5) nel registro *WaitTime*. In caso contrario il PCD chiamerà il blocco PB *LongTime* e trasferirà la costante *LongTimeValue* (15) nel registro *WaitTime*. Il registro *WaitTime* definisce la lunghezza della pausa tra due commutazioni di stato del lampeggiatore ("*Blink*"). Per assicurare l'inizializzazione del registro *WaitTime* durante un avviamento a freddo, il lampeggiatore deve essere posizionato dopo le due chiamate ai blocchi PB.

### Programma IL (Lista Istruzioni):

SAIA IL Editor [PB-IL] - [cligno.src]

File Edit Search View Project Online Tools Symbols Window Help

Group/Symbol	Type	Address/Value	Comment
ShortTimeValue	Constant	5	0.5 s
LongTimeValue	Constant	15	1.5 s
Tempo	Timer		

System Global cligno

```

COB      0
          0
STH      Condition           ;IF (Condition is high)
CPB      L ShortTime        ; THEN Call PB ShortTime
CPB      H LongTime         ; ELSE Call PB LongTime
ECOB

PB       ShortTime
STL     Tempo               ;IF tempo is low
LD      Tempo               ; load it with a short value
        ShortTimeValue
COM     Output              ; invert the output
EPB

PB       LongTime
STL     Tempo               ;IF tempo is low
LD      Tempo               ; load it with a long value
        LongTimeValue
COM     Output              ; invert the output
EPB

```

## 5.4 Blocchi Funzione (da FB 0 a 999)

### 5.4.1 Definizione

I Blocchi Funzione (FB) sono molto simili ai PB. Analogamente ai PB, i Blocchi Funzione contengono parti di programma che possono essere richiamati da altri blocchi. Anche in questo caso le chiamate possono essere condizionate o incondizionate.

L'unica differenza consiste nel fatto che è possibile effettuare una chiamata ad un Blocco FB con parametri, mentre per i PB questo non è permesso.

I blocchi FB offrono una soluzione ideale per sviluppare librerie di programmi utilizzabili per progetti diversi, consentendo così di ridurre i tempi di messa in servizio. La chiamata ad un FB con parametri può solo essere effettuata solamente da un programma IL.

Le chiamate a blocchi funzione possono essere nidificate fino ad un massimo di 7 livelli. Oltre i 7 livelli di nidificazione, il PCD effettuerà una chiamata al blocco XOB 10.

### 5.4.2 Esempio di chiamata funzione

L'esempio che segue si riferisce ad un FB che fa lampeggiare un'uscita.

L'FB viene richiamato due volte. La prima volta fa lampeggiare l'uscita 64 ogni 1,5 secondi, mentre la seconda volta fa lampeggiare l'uscita 65 ogni 3 secondi.

```

      FB      1          ;Inizio FB

tempo DEF  = 1          ;[T]   Indirizzo del temporizzatore
delay DEF  = 2          ;[W]   Pausa tra due commutazioni del lampeggiatore
blinker DEF = 3          ;[O,F]  Indirizzo del lampeggiatore

      STL    =tempo      ; se lo stato del temporizzatore è basso
      LDL    =tempo      ; inizializza il temporizzatore con parametro =2
              =delay
      COM    =blinker    ; commuta parametro =3
      EFB

      COB    0
              0

      CFB    1          ; Prima chiamata FB
              T 1
              15
              O 64

      CFB    1          ; Seconda chiamata FB
              T 2
              30
              O 65
      ECOB
  
```

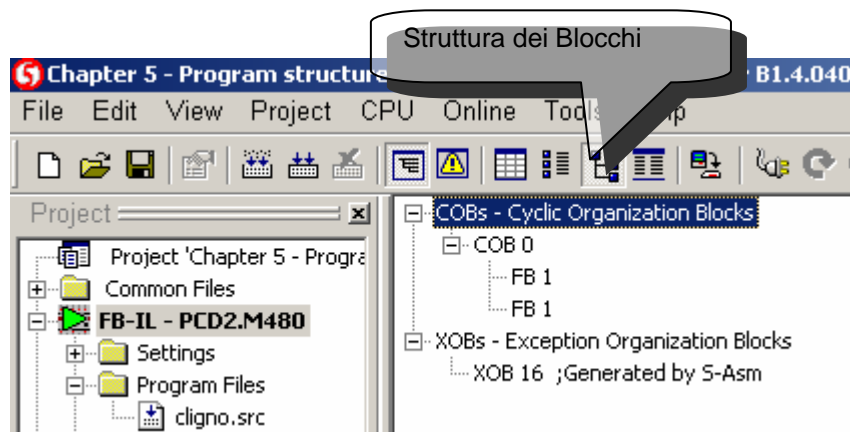
## 5.5 Visualizzazione della Struttura e Percorso di Esecuzione

Dopo aver costruito il programma, è possibile visualizzare la struttura dei Blocchi che costituiscono il programma stesso.

Fare clic sull'icona "Visualizza Struttura Blocchi" nella barra degli strumenti del Project Manager.

Verrà visualizzata la struttura che indica quali sono i COB che chiamano i vari PB, FB, o SB.

La figura seguente si riferisce all'esempio del FB descritto nella pagina precedente. Si può vedere infatti che FB1 è richiamato da COB 0 due volte.



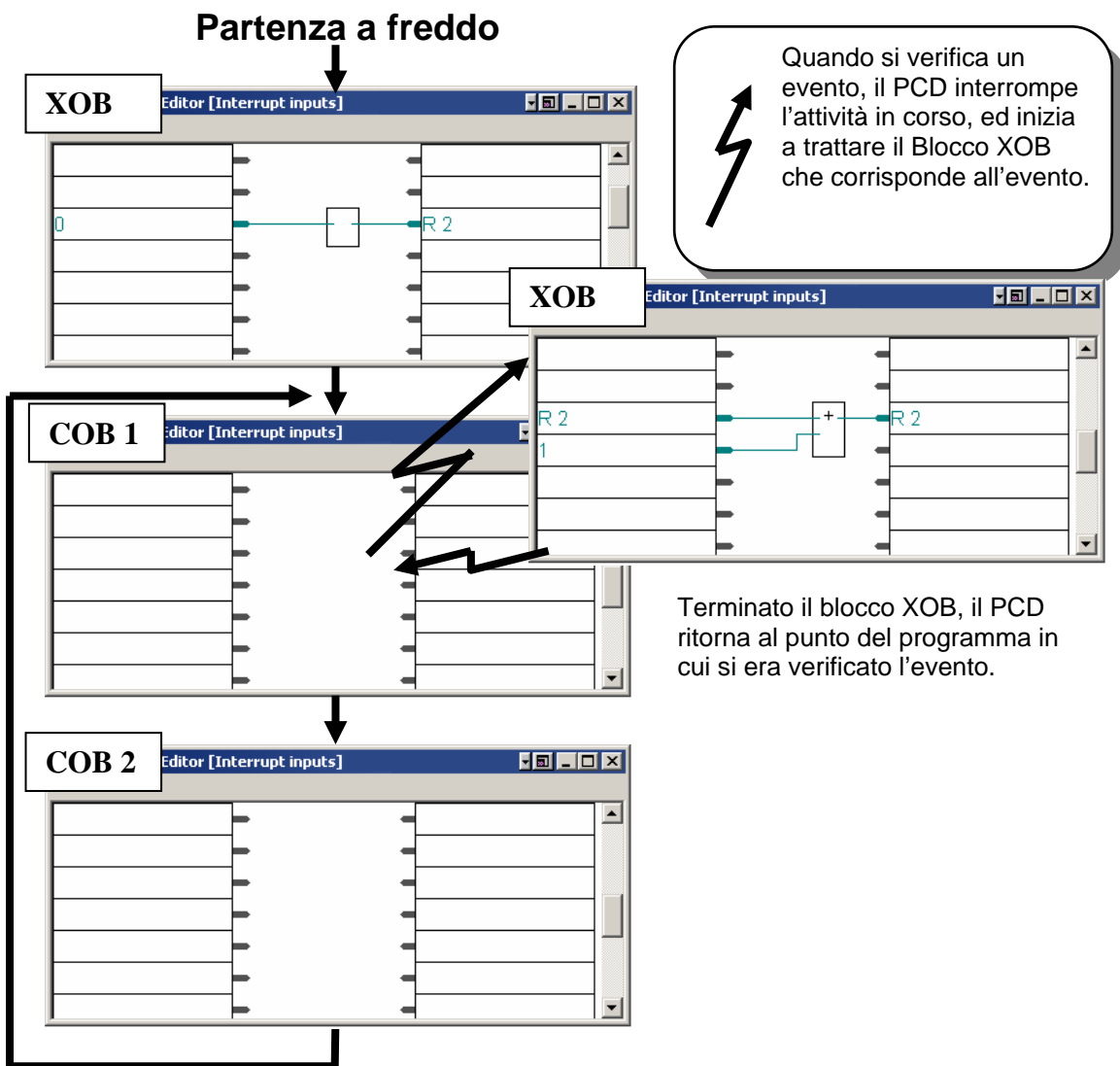
## 5.6 Blocchi ad Organizzazione Esclusiva (XOB)

### 5.6.1 Definizione

I blocchi ad organizzazione esclusiva sono programmi chiamati automaticamente non appena si verifica un evento particolare. Ciascun evento hardware o software è collegato ad un XOB. Questi eventi non possono essere modificati dall'utente. Tuttavia, l'utente è libero di programmare l'azione da intraprendere all'interno di ciascun XOB.

Esempio:

All'accensione, il PLC deve azzerare un registro utilizzato per conteggiare gli impulsi ad una frequenza massima di 1 kHz. Non è necessario utilizzare alcun programma speciale all'interno dei COB!



**Esempio:**

Avviare il PCD, rimuovere la batteria e verificare che si accenda il LED di errore. Se il programma prevede il blocco XOB 2 (vedere tabella nella pagina seguente), il LED non si accende, ma viene eseguito il blocco XOB 2.

**5.6.2 Elenco di tutti i blocchi XOB della famiglia PCD**

<b>XOB</b>	<b>Descrizione</b>	<b>Priorità</b>
0	Caduta di alimentazione nel rack principale (PCD6) o WatchDog (PCD1/2)	4
1	Caduta di alimentazione nel rack di estensione (PCD 6 )	2
2	Batteria scarica	2
4	Errore di parità sul Bus di I/O (PCD6)	1
5	Nessuna risposta dal modulo di I/O (PCD4/6)	1
7	Sovraccarico sistema dovuto ad eventi multipli	3
8	Codice istruzioni non valida	4
9	Troppi rami attivi (Graftec)	1
10	Superamento livelli di nidificazione PB/FB	1
11	Superamento tempo supervisione COB	3
12	Superamento capacità registro indice	1
13	Attivazione flag di errore	1
14	XOB Interruzione ciclica	3
15	XOB Interruzione ciclica	3
16	Partenza a freddo (eseguito all'accensione)	4
17	Richiesta interruzione tramite S-Bus	3
18	Richiesta interruzione tramite S-Bus	3
19	Richiesta interruzione tramite S-Bus	3
20	Ingresso di Interrupt INB1	3
25	Ingresso di Interrupt INB2	3
30	Nessuna connessione con RIO	1

Se si verifica un errore e non è stato programmato il blocco XOB corrispondente, si accende il LED di errore sul pannello frontale del PCD, e il programma utente prosegue normalmente.

Se si verifica un errore ed è stato programmato il blocco XOB corrispondente, il LED di errore sul pannello frontale del PCD rimarrà spento e verrà richiamata la routine esclusiva.

Un meccanismo di gestione priorità permette di elaborare i blocchi XOB più importanti con una priorità più elevata. Il livello di priorità più elevato è il livello 4.



### 5.6.3 Utilizzo dei Blocchi XOB

Supporto per individuare gli errori nella configurazione del programma:

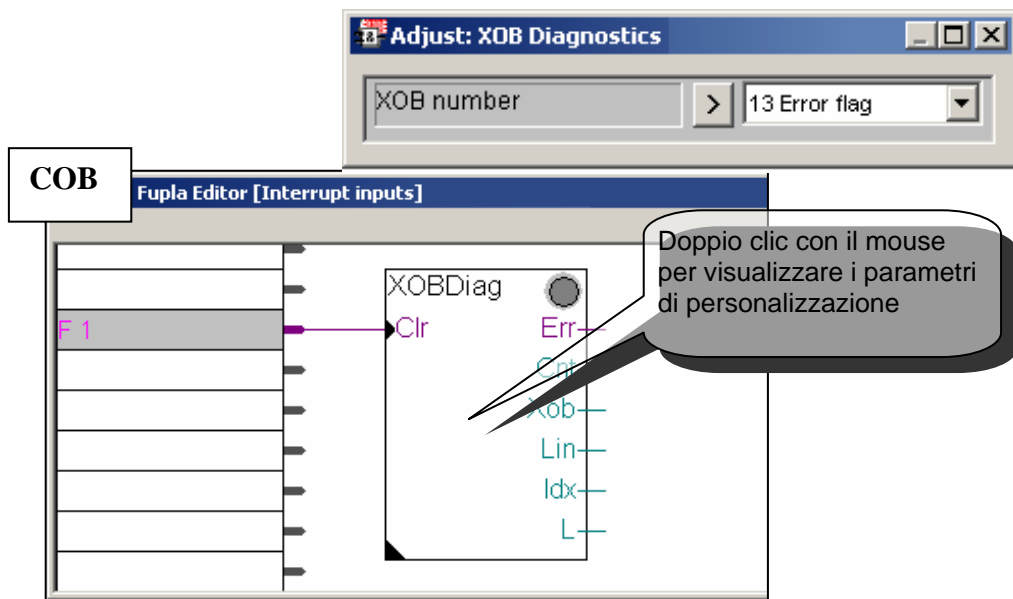
- Errori negli indirizzi dei moduli
- Superamento dei 7 livelli di nidificazione di programma
- Più di 32 transizioni attive in una struttura Graftec
- Loop infinito
- Errore in una operazione matematica
- Errori di comunicazione

#### Esempio con Fupla:

Utilizzo di tutti gli strumenti disponibili per la localizzazione sistematica degli errori nel programma utente.

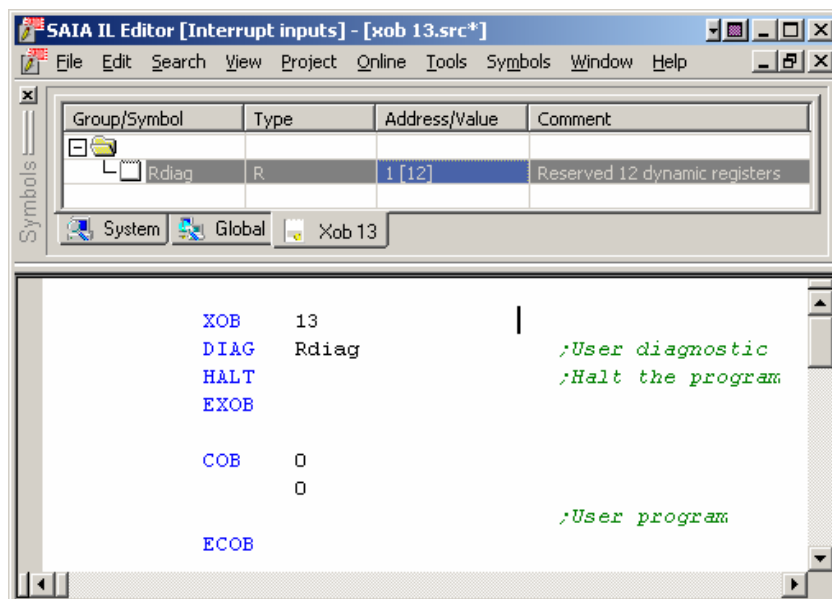
In Fupla non è necessario creare dei blocchi XOB. Questi vengono aggiunti automaticamente dalla funzione: *Special, Diagnostic XOB (Speciale, XOB Diagnostico)*.

Sono disponibili informazioni diagnostiche sulle uscite dalle funzioni, sul contatore degli errori, sul numero dell'XOB, sul numero di riga del programma ...



#### Esempio con Lista Istruzioni:

La diagnostica dei programmi scritti in IL fornisce le stesse informazioni sopra indicate nei registri Rdiag + 0 ... +12.



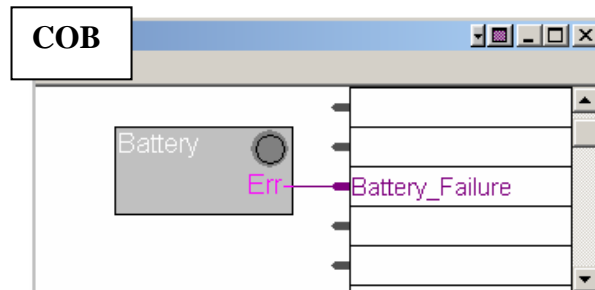
Manutenzione del PCD:

- Controllo delle batterie (devono essere sostituite ogni 3-4 anni)

#### Esempio con Fupla:

Con Fupla non è necessario creare un blocco XOB 2. Questo verrà aggiunto automaticamente dalla funzione: *Special, Battery (Speciale, Batteria)*

L'uscita Battery\_Failure sarà alta quando si verifica un qualsiasi problema legato alla batteria.



#### Esempio con Lista Istruzioni:

In presenza di una anomalia sulla batteria del PCD, si accende la lampada "battery" sul pannello frontale del PCD e verrà quindi chiamato automaticamente, ad intervalli regolari, il blocco XOB 2.

Nell'esempio rappresentato sotto, il blocco XOB 2 carica un temporizzatore con un ritardo di 1 secondo. Dal momento che il blocco ad organizzazione esclusiva viene richiamato ad intervalli regolari, il temporizzatore verrà inizializzato frequentemente e quindi non avrà la possibilità di raggiungere il valore zero. Lo stato binario di questo temporizzatore sarà quindi alto in caso di guasto della batteria, e ritornerà al livello basso circa un secondo dopo la sostituzione della batteria.

Group/Symbol	Type	Address/Value	Comment
Battery_Failure	Timer	1	

```

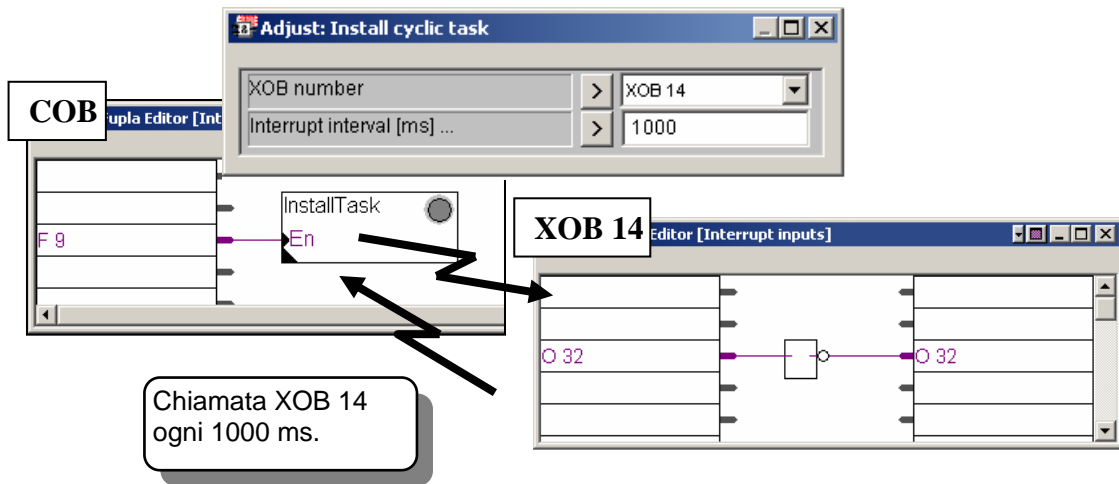
XOB 2 ;Alarm Battery
LD Battery_Failure
k 10
EXOB
COB 0 ;User program
0
STH Battery_Failure
ECOB
  
```

Controllo di eventi speciali o reazioni molto veloci a segnali esterni:

- Ingressi di Interrupt
- Interruzione occasionale del programma
- Interruzione del programma alla ricezione di un telegramma
- Partenza a freddo. Valori iniziali

**Esempio con Fupla:**

Emissione di impulsi su una uscita digitale. Utilizzare le funzioni *Special*, *Install cyclic task* e *Binary*, *Direct transfer* (*Speciale*, *Installa task ciclici e binari*, *Trasferimento Diretto*).



**Esempio con Lista Istruzioni:**

```

XOB      16
SYSWR   4014      ; Inizializza XOB 14
                    ; con un interrupt di 1000 ms

EXOB

COB      0
          0
                    ; Programma utente

ECOB

XOB      14      ; Interrupt ciclico
COM      O 32    ; con commutazione dell'uscita O 32
EXOB
    
```

### 5.6.4 Tabella Storica

Nella *Tabella Storica (PCD History Table)* vengono registrati tutti gli errori che si verificano a livello di hardware e di software.

La tabella viene aggiornata anche se non sono stati programmati i blocchi XOB.

Per consultare la tabella storica fare clic sul pulsante *Configuratore In Linea* oppure selezionare in sequenza le voci di menu *Tools, Online Configurator (Strumenti, Configuratore In Linea)*.



Configuratore  
In Linea

Reason	Address	Time	Date
>7 CALL LEVELS	30	14:09:43	06/01/2003
>7 CALL LEVELS	30	14:09:43	06/01/2003
>7 CALL LEVELS	30	14:09:43	06/01/2003
>7 CALL LEVELS	30	14:09:43	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
>> >7 CALL LEVELS	30	14:09:44	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
>7 CALL LEVELS	30	14:09:44	06/01/2003
BATT FAIL 816	0	14:09:43	06/01/2003
IR OVERFLOW 0	0	12:00:00	06/01/2003
ERROR FLAG 772	6	14:09:44	06/01/2003

Help

Clear History

Close

Data e ora  
Linea di programma  
Contatore errori  
Testo  
Ultimo errore

#### Note:

- Ogni CPU ha una propria tabella storica
- La riga BATT FAIL esiste solo sulla CPU 0
- La riga viene indicata solo se l'errore può essere assegnato ad una riga di programma. In caso contrario viene visualizzata in esadecimale
- Il blocco XOB 0 viene chiamato solo se è stato programmato.

## 5.6.5 Descrizione degli XOB

### **XOB 0: Caduta di tensione nel rack principale**

Il controllo della tensione nel modulo di alimentazione del rack principale ha rilevato una eccessiva caduta di tensione.

In questo caso, tutte le uscite vengono resettate nel modo seguente:

- Per il PCD4: immediatamente
- Per il PCD6: dopo 1,5 ms.

Viene richiamato il blocco XOB 0 e tutte le CPU vengono poste in stato HALT.

Il tempo che intercorre tra il momento in cui viene chiamato il blocco XOB 0 e il momento in cui le CPU vengono poste in stato HALT è di circa 5 ms. Durante questo tempo, il blocco XOB 0 continua l'elaborazione, in modo da poter salvare i dati.

### **XOB 1: Caduta di di tensione nel rack di espansione (PCD6)**

Il controllo della tensione nel modulo di alimentazione di un rack di espansione (PCD 2 o PCD6) ha rilevato una eccessiva caduta di tensione.

In questo caso tutte le uscite del rack di espansione vengono resettate entro 2ms e viene richiamato il blocco XOB 1.

Se le uscite di questo rack di espansione "morto" continuano ad essere gestite (attivate, disattivate o interrogate) dal programma utente in una qualsiasi CPU, verranno richiamati anche i blocchi XOB 4 e/o XOB 5.

### **XOB 2: Batteria guasta o scarica**

Batteria scarica, guasta o assente.

Le informazioni negli indicatori (flag) e registri non volatili, oppure il programma utente in RAM così come l'orologio hardware potrebbero risultare alterati. La segnalazione di batteria scarica potrebbe essere indicata anche dopo un periodo prolungato di non-utilizzo del PCD (più di 2 mesi senza alimentazione); in questo caso, tuttavia, non si avrebbe alcuna perdita di dati.

Gli stessi sintomi potrebbero essere indicati anche in caso di PCD nuovo, mai utilizzato in precedenza.

### **XOB 4 : Errore di parità sul bus indirizzi ( PCD6)**

Il blocco XOB 4 può essere chiamato solo se il PCD è dotato di rack di espansione.

Il circuito di controllo del bus indirizzi ha segnalato un errore di parità. Questo può essere causato da un guasto sul cavo, da un difetto sul rack di espansione o su un modulo di espansione bus, oppure semplicemente dal fatto che il rack di espansione indirizzato non è presente. In caso di guasto, potrebbe essere visualizzato un elemento errato.

### **XOB 5: Nessuna risposta dal modulo I/O (PCD4/6)**

I moduli di ingresso e uscita del PCD devono inviare un segnale di risposta alla CPU che li ha indirizzati. Se questo segnale non viene inviato, viene chiamato il blocco XOB 5. Questo avviene generalmente quando il modulo non è presente, ma può avvenire anche in caso di decodifica errata dell'indirizzo sul modulo.

In caso di modulo PCD4 con soli 8 elementi, il blocco XOB 5 non viene chiamato se viene indirizzato uno degli elementi assenti, dal momento che questo indirizzo è sempre decodificato e viene sempre inviato il segnale di risposta. Per quanto riguarda il PCD6, questo segnale viene inviato in risposta solo dai nuovi moduli PCD6. La maggior parte dei moduli PCA2 utilizzati finora non provocano la chiamata al blocco XOB 5, anche quando sono assenti.

**XOB 7: Sovraccarico di sistema**

Il meccanismo di attesa per i blocchi XOB con livelli di priorità 2 o 3 è sovraccaricato. Se un XOB di livello 2 o 3 viene elaborato nello stesso istante di un XOB con priorità più elevata (livello 4), il blocco XOB con priorità più bassa viene posto in attesa finché non è terminata l'elaborazione del blocco XOB con priorità più alta. Quando la coda di attesa è piena, viene chiamato il blocco XOB 7.

**XOB 8: Codice istruzioni non valido**

La CPU ha rilevato un codice istruzione non valido. Se i programmi o le routine scritti, vengono assemblati, linkati e caricati nel PCD, non si possono verificare errori nei codici delle istruzioni, in quanto il programma viene controllato dettagliatamente sia dall'editore IL (S Edit) che dall'assemblatore. Tuttavia, se il programma utente viene successivamente modificato utilizzando direttamente il programma Debug oppure utilizzando l'unità di programmazione portatile, esiste la possibilità di introdurre degli errori che potrebbero portare alla chiamata del modulo XOB 8.

Gli errori più frequentemente introdotti in questo modo sono: chiamata a blocchi non esistenti; istruzione di fine blocco mancante; salti di programma alla seconda riga di una istruzione multi-riga; salto da un blocco all'interno di un altro blocco ecc.

**XOB 9: Troppi rami attivi (GRAFTEC)**

Sono stati attivati simultaneamente più di 32 rami Graftec in un Blocco Sequenziale (SB). Ovviamente, si possono programmare più di 32 rami paralleli in un singolo SB, tuttavia, solo 32 di questi possono essere eseguiti simultaneamente.

**XOB 10: Più di 7 chiamate nidificate a PB/FB**

I blocchi PB e FB possono essere nidificati fino a 7 livelli di profondità. Una ulteriore chiamata, (chiamata all'ottavo livello) determina l'esecuzione dell'XOB 10. La chiamata all'ottavo livello non viene eseguita.

**XOB 11: Superamento tempo di supervisione COB**

Se la seconda linea di una istruzione COB indica un tempo di supervisione (in 1/100 di secondo) e il tempo di elaborazione del COB supera questa durata definita, viene chiamato il blocco XOB11. Il tempo di elaborazione COB è il tempo che può intercorrere tra le istruzioni COB ed ECOB. Lo scopo originale di questo tempo di supervisione era quello di poter scoprire immediatamente e quindi rimuovere eventuali blocchi o ritardi nel programma utente derivanti da errori di programmazione (loop di attesa, loop di conteggio troppo lunghi). In altri termini, si tratta di un "watchdog software". Come accennato in precedenza, i loop di attesa e conteggio (salti di programma) non sono consigliati. Questo minimizza la possibilità di blocco dei programmi utente. In ogni caso, anche nei programmi strutturati correttamente, si possono programmare uno o più COB con calcoli matematici molto lunghi ecc. che determinano lunghi tempi di esecuzione, per cui altri blocchi contenenti solo funzioni di supervisione e controllo possono essere ritardati. Se scade il tempo di supervisione definito per questo lungo programma di calcolo, il COB viene abbandonato e il programma prosegue a partire dall'inizio del COB successivo. Il "punto di abbandono" viene automaticamente memorizzato insieme allo stato dell'ACCU.

Quando viene nuovamente richiamato il COB originale, questo continua a partire dall'indirizzo di abbandono +1. Se si utilizza questa tecnica, non si dovrebbe programmare il blocco XOB 11, altrimenti si perderebbe tempo quando il timeout non è causato da un errore.

Nel paragrafo "Altre tecniche di programmazione" viene spiegata un'altra tecnica di programmazione (timeslice).

**XOB 12: Superamento capacità del registro indice**

La dimensione del Registro Indice è di 13 bit (da 0 a 8191). Questo è sufficiente per fare riferimento agli indirizzi di tutti gli elementi. Se un programma contiene un elemento indicizzato che non rientra nel suo campo di indirizzi, viene chiamato il blocco XOB 12.

Ad esempio, viene utilizzato come riferimento la Flag Indicizzata 8000 e il Registro Indice contiene 500, per cui l'indirizzo risultante risulterebbe 8500, che è al di fuori del campo indirizzi delle Flag 0 - 8191.

**XOB 13: Attivazione flag di ERRORE**

Molte istruzioni del repertorio istruzioni PCD possono attivare la flag di errore; vedere "Manuale delle Istruzioni, Elementi FLAG.

Nel caso in cui si verifichi un errore, oltre ad attivare la flag di Errore, viene anche chiamato il blocco XOB 13 che consente di effettuare gli aggiustamenti del caso (allarme, messaggio di errore su stampante, ecc.). Il blocco XOB 13 viene sempre chiamato quando si attiva la flag di errore, indipendentemente dal fatto che l'errore stesso sia stato causato da un calcolo, da un trasferimento dati o errori di comunicazione. Se per la flag di Errore è richiesta una diagnosi più precisa, si può chiamare in modo condizionato un blocco PB (o FB) dopo ogni istruzione in grado di attivare la flag di Errore.

Esempio:

```
....
DIV R 500 ; valore 1
R 520      ; valore 2
R 550      ; risultato
R 551      ; resto
CPB E 73   ; se errore chiama PB 73 ....
....
PB 73      ; Divisione per zero
SET O 99
INC C 1591
EPB
...
```

Il blocco PB 73 viene chiamato dopo una divisione per zero ed attiva l'Uscita 99, che indica appunto una condizione di divisione per zero. Il contatore C 1591 conteggia il numero di volte in cui si verifica questo evento. Il superamento della capacità di un registro potrebbe, ad esempio, attivare l'uscita 98 e il contatore C1590 potrebbe conteggiare questo tipo di eventi.

Anche il blocco XOB 13 dovrebbe essere programmato, ma può essere lasciato vuoto. Se non è programmato, quando si attiva il flag di errore, si accende anche la lampada di errore sul pannello frontale della CPU e questo potrebbe non essere desiderato.

**IMPORTANTE:**

La flag di Errore e le altre flag di stato aritmetiche (Positivo, Negativo, Zero) vengono attivati in caso di eventi o stati particolari, e se lo si desidera, possono essere elaborati immediatamente, dal momento che queste flag di stato si riferiscono sempre all'ultima istruzione eseguita che li può influenzare.

Ad esempio, se dopo la divisione per zero indicata nel caso precedente venisse eseguita una addizione corretta, la flag di errore verrebbe disattivata.

**XOB 14, 15: XOB di interruzione ciclica**

I blocchi XOB 14 e 15 vengono richiamati periodicamente con una frequenza variabile da 10 ms a 1000 s. Tale frequenza può essere definita con l'istruzione SYSWR.

**XOB 16: Partenza a freddo**

XOB 16 è il blocco che gestisce la partenza a freddo, viene elaborato all'accensione del PCD, oppure quando quest'ultimo riceve un comando di ripartenza a freddo dall'apparecchio di programmazione. Il blocco XOB 16 può essere utilizzato per inizializzare tutti i tipi di informazioni prima di elaborare il programma. Una volta terminato il blocco XOB 16, il programma inizierà ad elaborare i COB in ordine crescente di numero, senza mai ritornare al blocco XOB 16 stesso.

XOB 16 non può essere riavviato dal programma utente. Nel caso in cui si debba eseguire una particolare azione sia da parte di un COB, che durante l'inizializzazione, tale azione deve essere scritta in un blocco PB o FB richiamabile sia da XOB 16 che dal COB in questione.

**XOB 17, 18, 19: Richiesta interruzione tramite S-Bus**

Questi tre XOB possono essere utilizzati come routine di interrupt. E' possibile iniziare ad elaborarli attraverso il bus di comunicazione S-Bus. A questo scopo si può utilizzare l'interruzione SYSWR oppure la funzione *Fupla Special, execute XOB* (Speciale, esecuzione XOB).

**XOB20 & XOB 25: Gestione ingressi di interrupt**

Il blocco XOB 20 (o 25) viene chiamato quando sull'ingresso di interrupt INB1 (o INB2) del PCD2 viene rilevato un fronte di salita (per ulteriori dettagli fare riferimento al manuale hardware PCD2).

**XOB 30: Perdita del collegamento master-slave con moduli RIO**

Il collegamento viene verificato dopo ogni messaggio inviato dalla stazione master ad una stazione slave. Se la verifica dà esito negativo la CPU master chiama il blocco XOB 30. Questo si verifica, ad esempio, quando una stazione viene scollegata dalla rete in online o viene spenta.

## 5.7 Blocchi Sequenziali (da SB 0 a 31, 96<sup>1</sup>)

I blocchi sequenziali sono rappresentati da un insieme di cosiddetti PASSI e TRANSIZIONI.

In un passo viene eseguita una parte del programma, mentre in una transizione si attende che si verifichi la condizione necessaria per proseguire con il passo successivo.

I blocchi sequenziali sono creati con un editore speciale. Tale editore è denominato GRAFTEC e i file prodotti hanno estensione \*.sfc. L'editore Graftec è spiegato nel prossimo capitolo. Esso rappresenta uno strumento eccellente quando si devono programmare attività che devono essere trattate in maniera sequenziale.

I blocchi sequenziali possono essere richiamati da un qualsiasi altro blocco.

---

<sup>1</sup> PCD2.M170/480, PCD4. M170 e PCD3



## 5.8 Tabella riassuntiva.

Servizio	Mezzo	Operando	Note
Cyclic Organization Block (Blocco a Organizzazione Ciclica)	COB	0...15	Minimo 1 COB per programma
Program Block (Blocco di Programma)	PB	0...299	Sotto programmi chiamati da un COB, PB,FB,SB or XOB
Function Block (Blocco Funzione)	FB	0...999	Funzione con parametri chiamata da un COB, PB,FB,SB o XOB
Sequential Block (Blocco Sequenziale)	SB	0...32 0...96 <sup>1</sup>	Sotto programmi sequenziali programs chiamati da un COB, PB o FB ( SB, XOB)
Step (Passo)	ST	0...1999 0...5999 <sup>1</sup>	
Transition (Transizione)	TR	0...1999 0...5999 <sup>1</sup>	

---

<sup>1</sup> PCD2.M170/480, PCD4. M170 e PCD3



## Indice

---

6	Programmazione Graftec	6-3
6.1	Introduzione	6-3
6.2	Blocchi Sequenziali (da SB 0 a 31, 96 )	6-4
6.3	Blocchi ad Organizzazione Ciclica	6-5
6.3.1	Programmi Ciclici	6-5
6.3.2	Tempo di Ciclo	6-5
6.4	Creazione di un nuovo file Graftec	6-6
6.4.1	Creazione di un nuovo progetto	6-6
6.4.2	Creazione di un nuovo file FUPLA o IL	6-6
6.4.3	Chiamata dell'SB da un COB	6-7
6.4.4	Creazione di un nuovo file Graftec	6-7
6.5	Organizzazione dei Blocchi SB	6-8
6.5.1	Navigatore Blocchi	6-8
6.5.2	Struttura Generale di un SB	6-8
6.5.3	Regole di evoluzione	6-9
6.5.4	Transizioni (da TR 0 a 1999 )	6-10
6.5.5	Passi (da ST 0 a 1999 )	6-11
6.6	Strutture Tipiche dei Blocchi Sequenziali.	6-12
6.6.1	Sequenza Semplice	6-12
6.6.2	Rami alternativi (OR)	6-12
6.6.3	Rami simultanei (AND)	6-12
6.6.4	Salto di una sequenza	6-12
6.6.5	Ripetizione di una sequenza	6-12
6.7	Scrittura di una Sequenza	6-13
6.7.1	Barra degli strumenti Graftec	6-13
6.7.2	Scrittura di una semplice sequenza	6-13
6.7.3	Tracciare una connessione	6-13
6.7.4	Scrittura di un Task Alternativo (OR)	6-14
6.7.5	Chiusura di un Task Alternativo	6-14
6.7.6	Scrittura di un task simultaneo (AND)	6-14
6.7.7	Chiusura di un Task simultaneo	6-14
6.7.8	Aggiunta di un commento	6-14
6.7.9	Inserimento di una sequenza	6-15
6.7.10	Cancellazione di una Sequenza	6-15
6.7.11	Copiare/Incollare una sequenza	6-16
6.8	Scrittura del primo Blocco Sequenziale	6-17
6.8.1	Apertura del file	6-17
6.8.2	Definizione della Struttura Base	6-17
6.8.3	Scelta dell'editore: IL o Fupla	6-18
6.8.4	Preparazione dei Simboli	6-19
6.8.5	Scrittura del programma	6-19

6.8.6	Come programmare una transizione	6-20
6.8.7	Utilizzo dei Temporizzatori in un SB	6-20
6.8.8	Ripetizione del passo e della transizione per tutto il tempo in cui l'impulso è disattivato	6-21
6.8.9	Decremento di un contatore	6-22
6.8.10	Diramazione alternativa	6-22
6.9	Costruzione (Build) e debug del programma	6-23
6.9.1	Finestra Messaggi	6-23
6.9.2	Strumenti Online	6-23
6.10	Come strutturare un programma GRAFTEC in pagine	6-24
6.10.1	Definizione di una pagina	6-24
6.10.2	Modifica di una pagina	6-25

## 6 Programmazione Graftec

---

### 6.1 Introduzione

L'Editore Graftec permette di creare programmi sequenziali usando i linguaggi Fupla o IL. Questo capitolo fornisce le informazioni basilari per utilizzare l'editore Graftec per procedere alla scrittura e modifica di programmi contenenti blocchi sequenziali (SB), passi (step) e transizioni (transition).

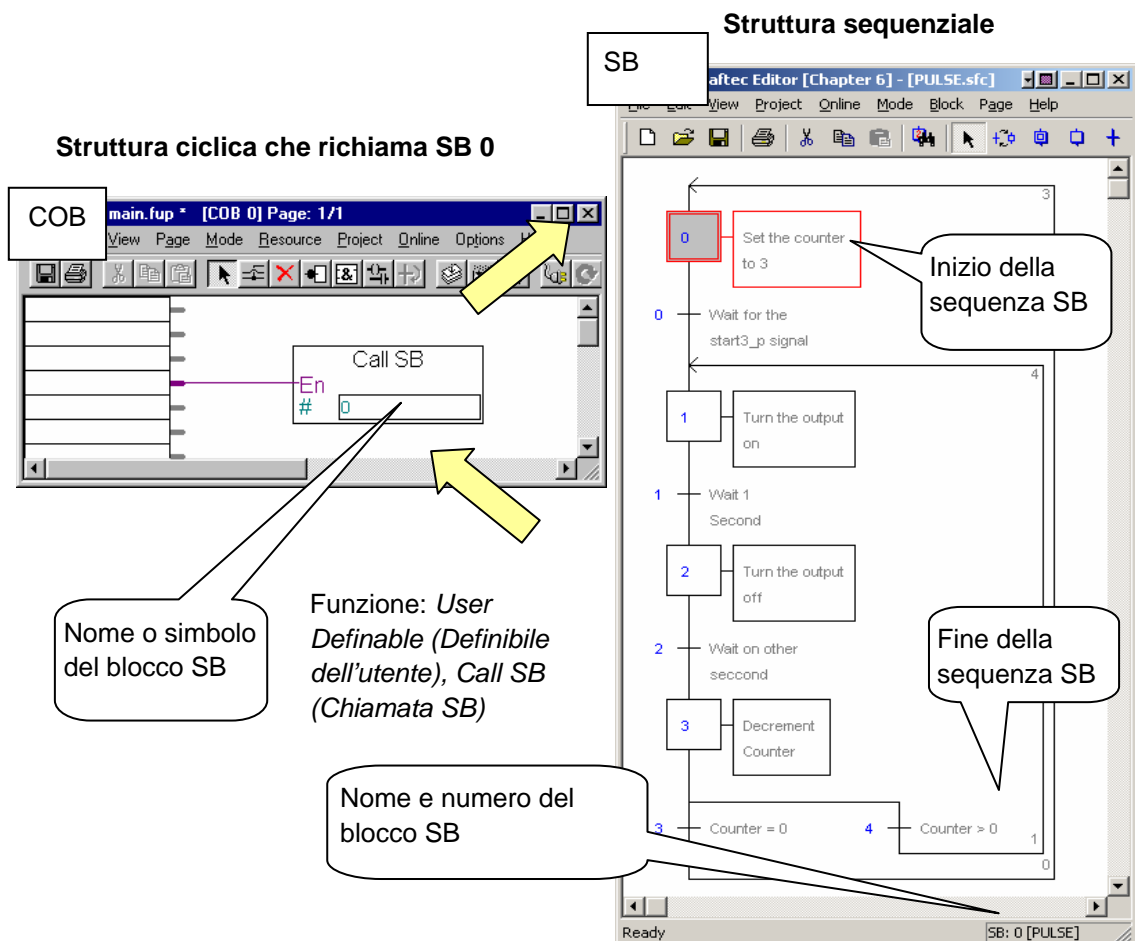
## 6.2 Blocchi Sequenziali (da SB 0 a 31, 96<sup>1</sup>)

Nelle pagine seguenti descriveremo una tecnica di strutturazione dei programmi denominata Graftec (o Schema Sequenziale delle Funzioni), che risulta particolarmente efficiente per i programmi sequenziali in cui è necessario attendere determinati eventi programmabili o esterni al controllore.

Poiché tali eventi hanno una durata indeterminata, non è possibile valutare il tempo di ciclo dei programmi sequenziali. E' perciò importante separare completamente i programmi ciclici dai programmi sequenziali.

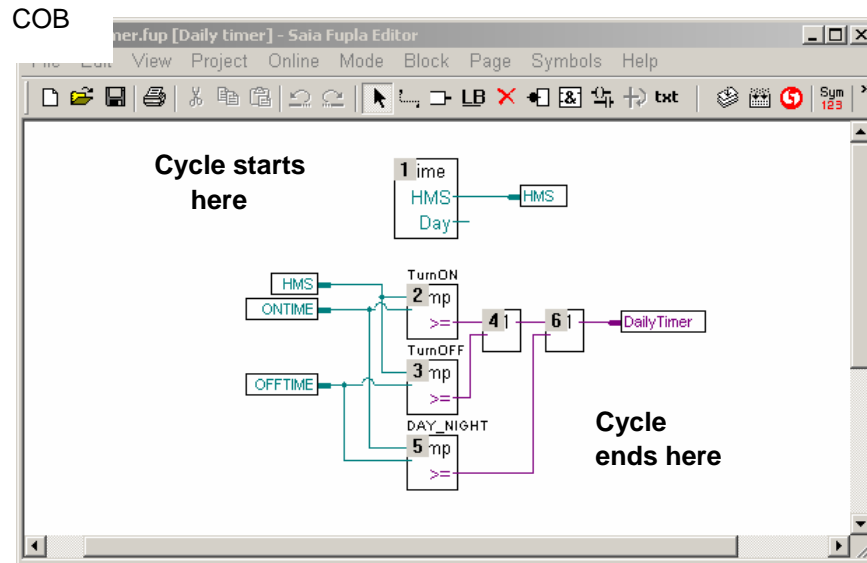
L'attesa di un evento sequenziale non deve mai bloccare l'esecuzione continua dei programmi ciclici. Per soddisfare questo requisito, i programmi sequenziali vengono inseriti all'interno di una delle 32 possibili strutture SB che possono essere richiamate in ciascun ciclo di programma.

Una caratteristica particolare dei blocchi SB è data dal fatto che, quando un programma sequenziale inserito all'interno del blocco SB è in attesa di un evento, il PCD ignora temporaneamente tale SB e prosegue con l'elaborazione dei programmi ciclici. La parte rimanente del blocco SB verrà quindi elaborata durante il ciclo di programma successivo.



<sup>1</sup> I nuovi PCD2/4.M170, PCD2.M480 e PCD3 supportano fino a 96 SB

## 6.3 Blocchi ad Organizzazione Ciclica



### 6.3.1 Programmi Ciclici

I programmi presentati in precedenza erano interamente ciclici: creati partendo da un elenco di funzioni o istruzioni grafiche elaborate una dopo l'altra dal PLC nel più breve tempo possibile, dall'inizio alla fine del programma stesso, e ritornando all'inizio per eseguire un nuovo ciclo.

### 6.3.2 Tempo di Ciclo

Il tempo richiesto per elaborare un ciclo di programma è fisso. Più precisamente, corrisponde alla somma dei tempi di esecuzione delle singole istruzioni e funzioni. Questo è quello che viene definito *tempo di ciclo* del programma.

Generalmente, il tempo di ciclo è dell'ordine di alcuni millisecondi. Se l'informazione presente su un ingresso digitale cambia stato, le uscite digitali che dipendono dallo stato dell'ingresso possono essere aggiornate con un tempo di rinfresco pressoché istantaneo. Il ritardo, è di fatto equivalente al tempo di ciclo.

I programmi ciclici appartengono a strutture di tipo: COB, PB, FB o XOB.

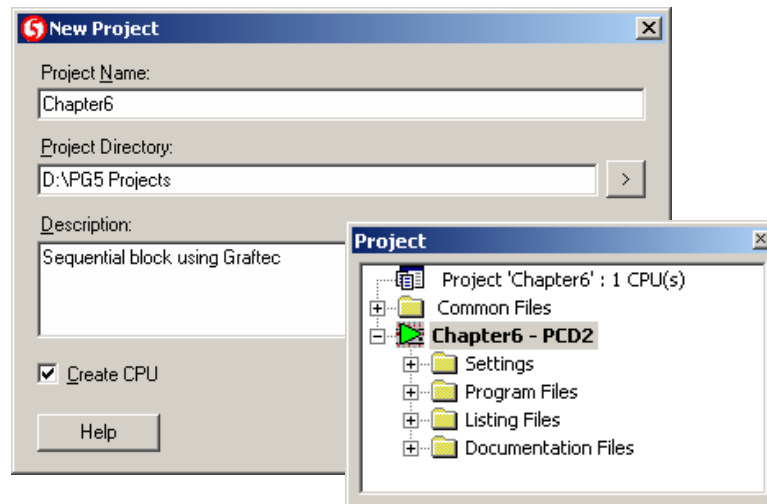
## 6.4 Creazione di un nuovo file Graftec

A titolo di esempio, consigliamo di creare un nuovo progetto in cui inseriremo i file necessari per la scrittura dei programmi Graftec:

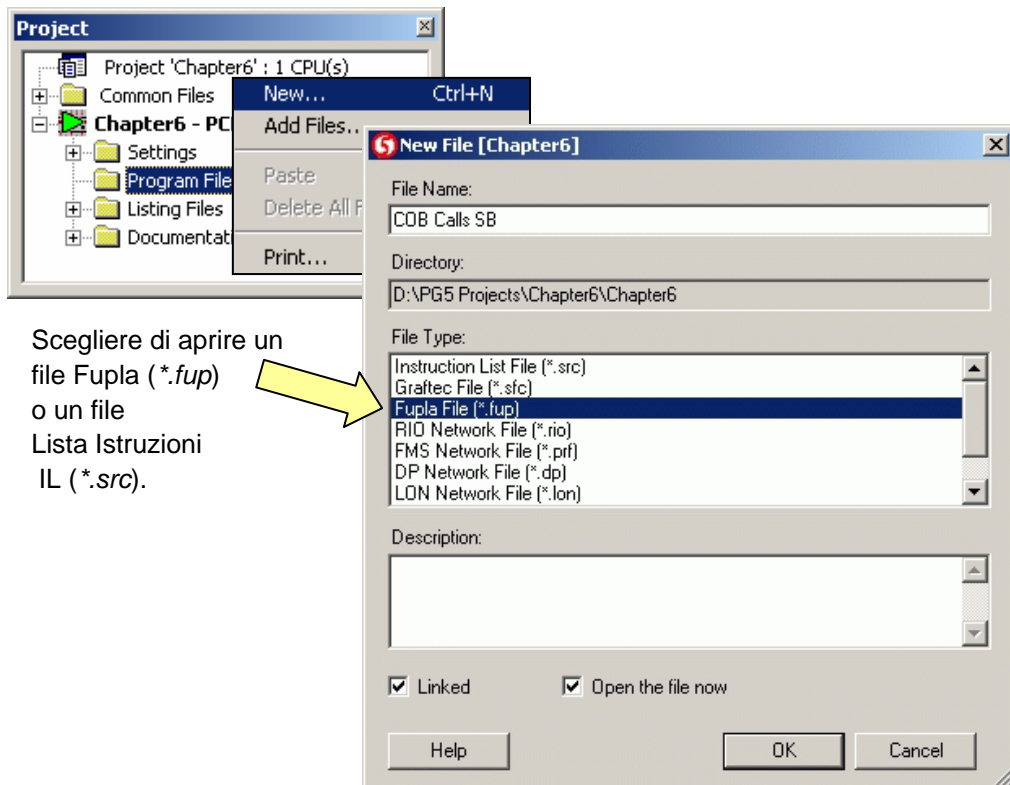
- Per la programmazione grafica, sarà necessario preparare un file Fupla ed un secondo file Graftec
- Per la programmazione in Lista Istruzioni, sarà invece necessario preparare un file IL ed un secondo file Graftec.

### 6.4.1 Creazione di un nuovo progetto

Dalla finestra SAIA Project Manager (Gestore Progetti SAIA), selezionare la voce di menu Project, New... (Progetto, Nuovo...) per creare il nuovo progetto.



### 6.4.2 Creazione di un nuovo file FUPLA o IL





### 6.4.3 Chiamata dell'SB da un COB

In base al modo con cui si intende scrivere il programma (Lista Istruzioni o FUPLA) si può richiamare il blocco SB utilizzando una istruzione *Call SB* (CSB) oppure la funzione Call SB. Non vi sono differenze sostanziali tra i due esempi. Aprire il file appena creato e scrivere il programma nel modo qui di seguito indicato.

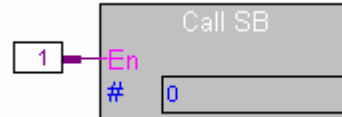
#### Programma in Lista Istruzioni:

```
COB    1; Inizio Blocco
        Ciclico
        0

CSB    0; chiama l'SB 0

ECOB   ; Fine Blocco Ciclico
```

#### Programma in FUPLA:



Fbox: *User definable (Definibile dall'utente), Call SB (Chiamata SB)*

### 6.4.4 Creazione di un nuovo file Graftec

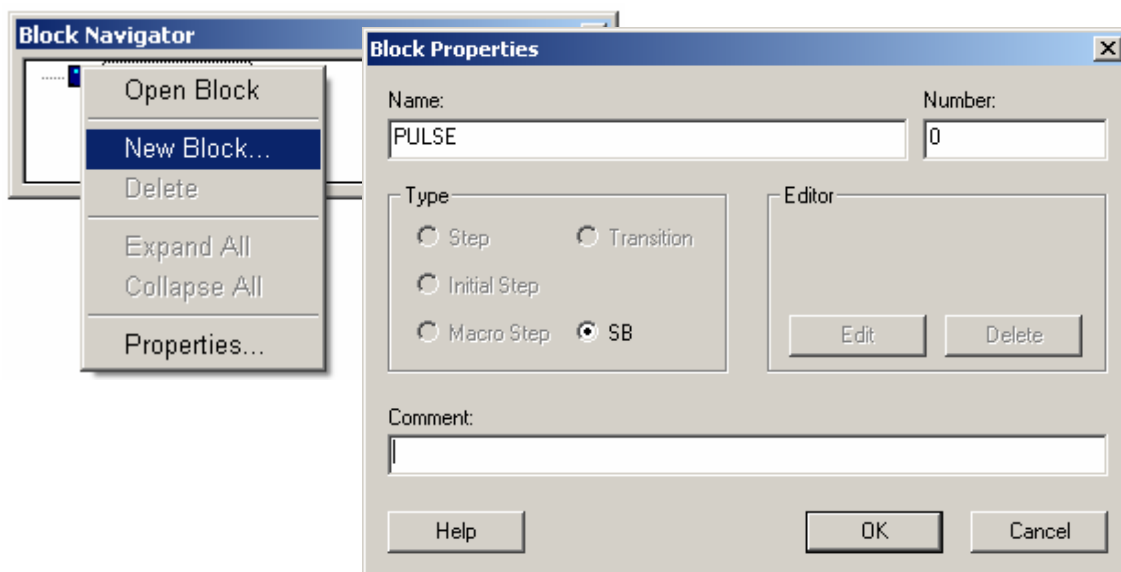
Scegliere di aprire un file Graftec (\*.sfc)

## 6.5 Organizzazione dei Blocchi SB

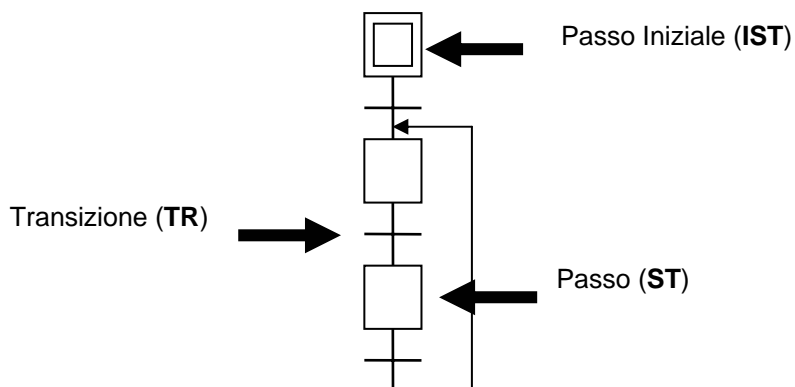
### 6.5.1 Navigatore Blocchi

Quando si crea un file Graftec, l'editore Graftec apre un Blocco Sequenziale 0 e predisporre un passo iniziale. Il blocco SB 0 viene quindi automaticamente inserito nell'elenco degli SB (*SB List*). Per accedere a questo elenco, aprire il file Graftec e selezionare il pulsante *Navigatore Blocchi*, oppure il menu *Block, Block Navigator (Blocchi, Navigatore Blocchi)*. In questo modo verrà visualizzato l'elenco dei blocchi SB nel file aperto. Ogni CPU può avere fino a 32 SB in un file, anche se è preferibile avere un file per ogni SB. Per visualizzare l'SB è sufficiente fare clic su *Properties (Proprietà)* avendo selezionato il numero di SB desiderato. (Non confondere questo numero con i numeri dei singoli elementi. Questo numero di riferisce all'intera stringa di elementi all'interno dell'SB). È inoltre possibile inserire un nome, per cui si raccomanda di assegnare un nome ad ogni SB. Questo faciliterà la navigazione attraverso i vari SB. Se non si inserisce un nome, l'elenco conterrà solo i numeri.

  
*Navigatore  
 Blocchi*



### 6.5.2 Struttura Generale di un SB

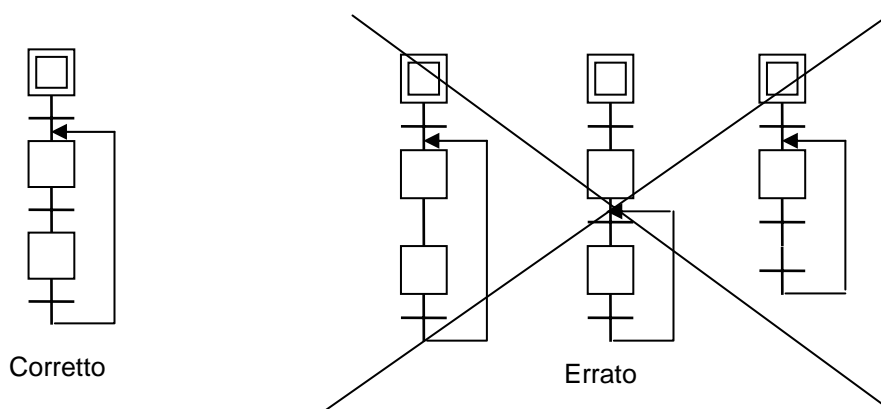


L'editore Graftec permette di scrivere la struttura dei blocchi sequenziali come sequenza di passi e transizioni, all'interno dei quali l'utente scrive il programma sotto forma di funzioni grafiche e elenco di istruzioni.

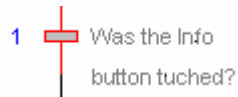
Un blocco sequenziale (SB) inizia con un passo iniziale, indicato con il simbolo di un doppio riquadro. Questo rappresenta l'inizio del programma, cioè il punto da cui il programma stesso inizierà quando viene chiamato per la prima volta il blocco (partenza a freddo).

### 6.5.3 Regole di evoluzione

I blocchi Sequenziali hanno una sintassi ben precisa. In particolare, un blocco sequenziale inizia sempre con un passo iniziale, quindi le transizioni e i passi devono susseguirsi in modo alternato. Non devono quindi mai essere scritti due passi o due transizioni collegati tra loro.



### 6.5.4 Transizioni (da TR 0 a 1999 <sup>1</sup>)



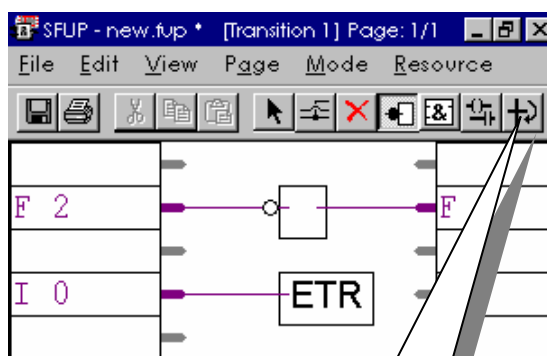
Inserire in una transizione le parti del programma che devono essere eseguite ripetutamente finché non si verifica una determinata situazione, ad esempio:

- Attesa di ricevere un carattere da una porta seriale
- Attesa di una condizione di tempo scaduto di un temporizzatore
- Attesa di una segnalazione di fine corsa

La transizione deve terminare sempre con un blocco *ETR*. La transizione viene ripetuta in continuazione se l'ingresso dell'FBox *ETR* è basso, o se l'ACCU è basso al termine di una transizione in Lista Istruzioni.

**Esempio:** La Flag 2 viene commutata ad ogni ciclo di programma finché l'ingresso 0 è alto.

#### Programma FUPLA:



Transizione.  
Qui viene richiamato l'FBox

#### Programma in Lista Istruzioni:

```

; Transition toggles Flag2 until
; Input 0 comes to high
COM F 2
STH I 0

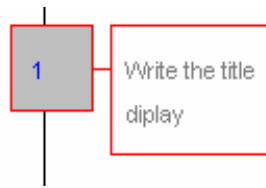
```

L'istruzione STH imposta l'ACCU(mulatore) allo stato alto, quando l'ingresso I0 è allo stato alto. La transizione viene considerata conclusa quando ACCU è alto al termine della transizione.

- Non è necessario inserire un programma in ogni transizione. Una transizione senza programma è sempre considerata vera e quindi verrà saltata.
- Transizioni scritte in Lista Istruzioni: l'ACCU è sempre alto all'inizio di una transizione o di un passo.
- Si possono utilizzare fino a 2000 passi e transizioni per costruire fino a 32 Blocchi Sequenziali.

<sup>1</sup> I nuovi PCD2/4.M170, PCD2.M480 e PCD3 supportano fino a 6000 ST/TR e 96 SB

### 6.5.5 Passi (da ST 0 a 1999 <sup>1</sup>)

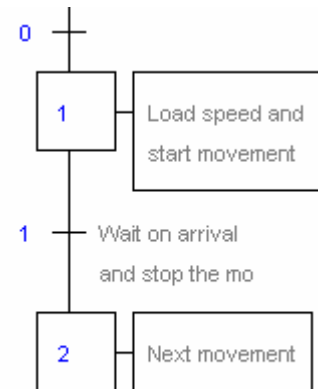


I passi contengono normalmente le parti di 'azione' di un programma, che vengono eseguite una volta sola quando si attiva la transizione precedente.

Supponiamo di voler comandare il movimento di un motore da A a B.

Tipicamente si dovrà prima impostare la velocità e la direzione del movimento. Quindi si dovrà iniziare il movimento. Entrambe le attività sono attività non ripetitive che possono essere inserite in un passo, dal momento che quest'ultimo viene eseguito una volta sola.

Una volta avviato il movimento, sarà necessario controllare il movimento stesso ed arrestare il motore quando questo raggiunge la destinazione B. Questo monitoraggio (ad esempio: lettura del fine corsa) deve essere effettuato in continuo, finché il motore non raggiunge la destinazione. Questo dovrebbe essere eseguito in una transizione, dal momento che le transizioni vengono eseguite ciclicamente.



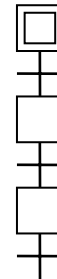
- Un passo senza programma va direttamente alla transizione successiva.
- Un passo viene trattato una volta sola! Non è ciclico!

<sup>1</sup> I nuovi PCD2/4.M170, PCD2.M480 e PCD3 supportano fino a 6000 ST/TR e 96 SB

**6.6 Strutture Tipiche dei Blocchi Sequenziali.**

**6.6.1 Sequenza Semplice**

La sequenza semplice comprende una alternanza di passi e transizioni. Non ci possono essere due passi o transizioni di seguito, nello stesso ramo.

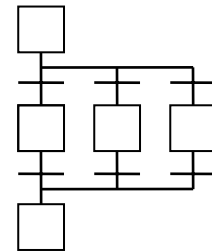


**6.6.2 Rami alternativi (OR)**

La ramificazione alternativa è la scelta di una sequenza tra varie possibilità.

Le transizioni vengono eseguite da sinistra a destra, e la prima transizione che presenta una condizione 'vera' determina quale sequenza deve essere elaborata. La ramificazione alternativa inizia sempre con un passo che si dirama in un certo numero di transizioni e termina con una inversione di tale struttura: un certo numero di transizioni si incanalano in un singolo passo.

L'editore Graftec supporta fino a 32 rami. Al di sopra dei 32 rami, il PCD richiama l'XOB 9 (vedere capitolo 5).

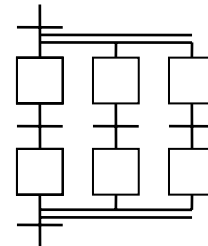


**6.6.3 Rami simultanei (AND)**

La ramificazione simultanea comprende un certo numero di sequenze che devono essere elaborate contemporaneamente.

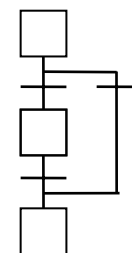
La ramificazione simultanea inizia sempre con una transizione che si dirama in un certo numero di passi e termina con una inversione di tale sequenza: un certo numero di passi si incanalano in una singola transizione di sincronizzazione.

L'editore Graftec supporta fino a 32 rami. Al di sopra dei 32 rami, il PCD richiama l'XOB 9 (vedere capitolo 5).



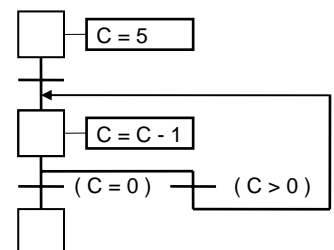
**6.6.4 Salto di una sequenza**

La ramificazione alternativa può essere utilizzata per saltare una sequenza, permettendo quindi l'elaborazione condizionale di tale sequenza.



**6.6.5 Ripetizione di una sequenza**

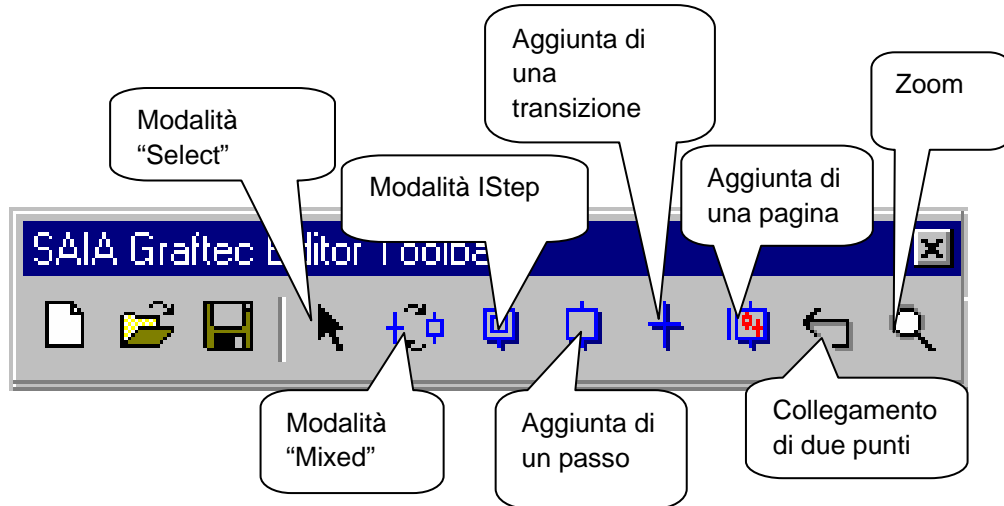
La ripetizione di una sequenza può essere effettuata anche con la diramazione alternativa. Ad esempio, un contatore potrebbe essere inizializzato con un numero di loop di programma. A questo punto si potrebbe inserire una semplice sequenza di lunghezza qualsiasi, decrementare il contatore e, se il contatore non ha ancora raggiunto il valore zero, il loop viene ripetuto.



## 6.7 Scrittura di una Sequenza

Quando si apre un nuovo file Graftec, viene visualizzato il passo iniziale. L'esecuzione di un SB inizia sempre da questo punto. Per poter aggiungere nuovi elementi allo schema, si può utilizzare indifferentemente la barra degli strumenti o la tastiera.

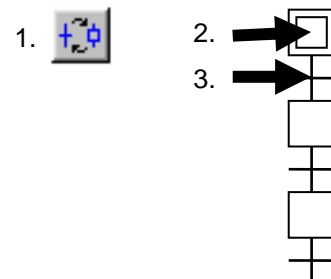
### 6.7.1 Barra degli strumenti Graftec



All'inizio di questo capitolo avevamo creato il file *Pulse.sfc*. Aprire questo file, selezionare l'elenco dei Blocchi SB e caricare il Blocco SB "OpenMainGate". Eseguire l'esercizio qui di seguito proposto.

### 6.7.2 Scrittura di una semplice sequenza

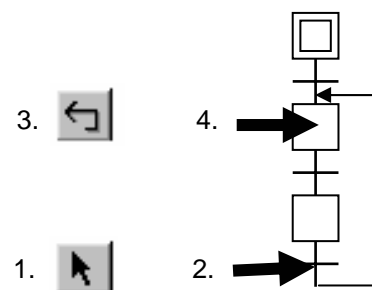
1. Selezionare la modalità *Mixed*
2. Posizionare il puntatore sotto il passo iniziale e fare clic con il pulsante sinistro del mouse.
3. Posizionare il puntatore sotto la nuova transizione e fare nuovamente clic con il pulsante sinistro del mouse.
4. Seguire questo formato.



### 6.7.3 Tracciare una connessione

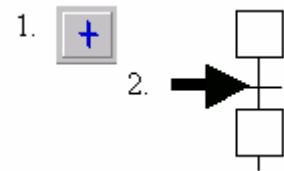
Una volta terminata la sequenza, termina anche il programma. Se si vuole riavviare il programma sarà necessario aggiungere un loop. Non è possibile tracciare una connessione tra due passi o tra due transizioni. Un loop inizia sempre su una transizione e termina su un passo.

1. Selezionare la modalità *Select*
2. Contrassegnare la transizione di inizio
3. Selezionare la modalità *Link*
4. Fare clic sul passo che si vuole collegare alla transizione.



### 6.7.4 Scrittura di un Task Alternativo (OR)

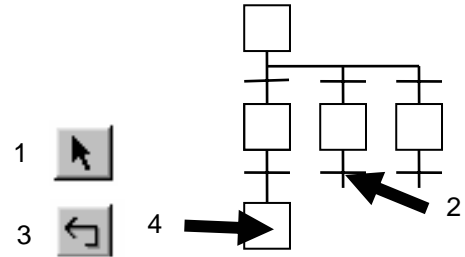
1. Selezionare la modalità *Transizione*
2. Fare clic su una transizione che sia già seguita da un passo
3. Ad ogni clic viene aggiunta una nuova transizione



### 6.7.5 Chiusura di un Task Alternativo

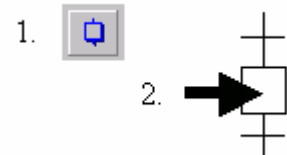
Per chiudere un task OR:

1. Selezionare la modalità *Select*
2. Contrassegnare la transizione che si desidera chiudere
3. Selezionare la modalità *Link*
4. Fare clic sul Passo che si desidera collegare.



### 6.7.6 Scrittura di un task simultaneo (AND)

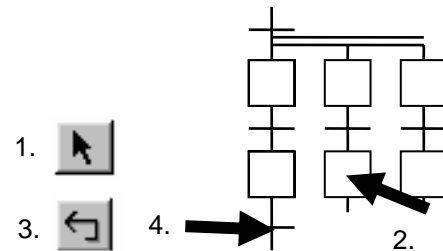
1. Selezionare la modalità *Passo*
2. Fare clic su un passo che sia già seguito da una transizione.
3. Ad ogni clic viene aggiunto un nuovo passo.



### 6.7.7 Chiusura di un Task simultaneo

Per ri-sincronizzare i task simultanei:

1. Selezionare la modalità *Select*
2. Contrassegnare il task che si desidera chiudere
3. Selezionare la modalità *Link*
4. Fare clic sulla transizione che si desidera collegare.

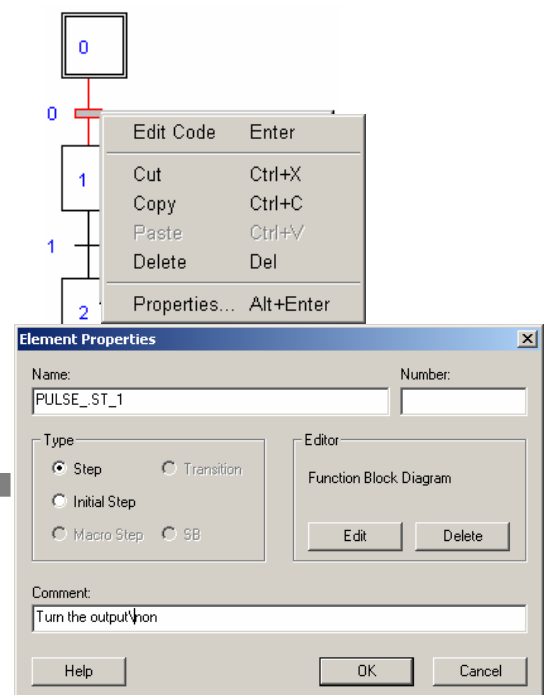
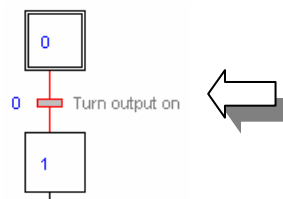


### 6.7.8 Aggiunta di un commento

1. Attivare la modalità *Selezione*
2. Fare clic su un elemento utilizzando il tasto destro del mouse. Scegliere *Properties.... (Proprietà)*
3. Inserire il commento all'interno del campo *Comment (Commento)*.

Nota:

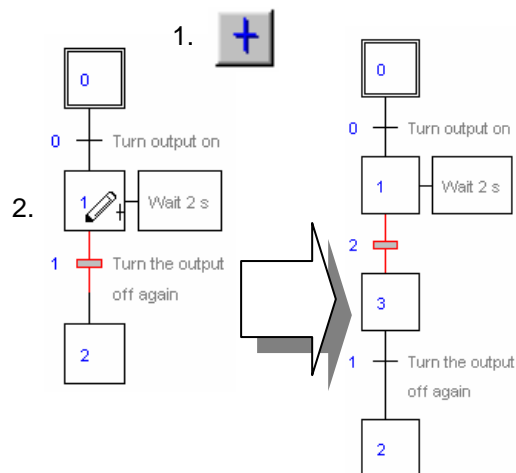
Per definire un commento su due righe, inserire la notazione“\n”.





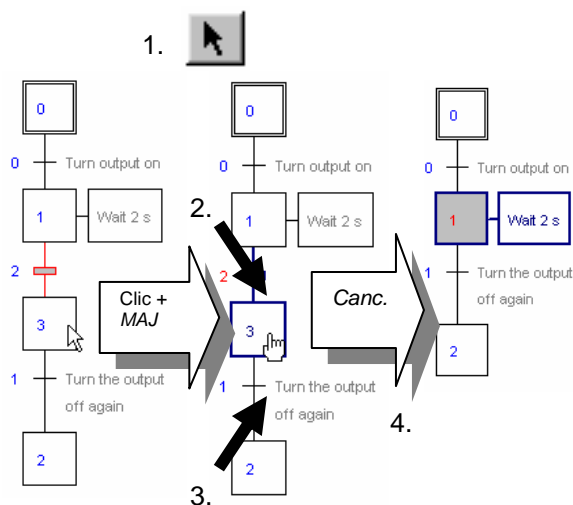
### 6.7.9 Inserimento di una sequenza

1. Selezionare la modalità *Transizione*
2. Fare clic su un passo che sia già seguito da una transizione.
3. L'Editore provvederà ad inserire una nuova Transizione ed un nuovo Passo.



### 6.7.10 Cancellazione di una Sequenza

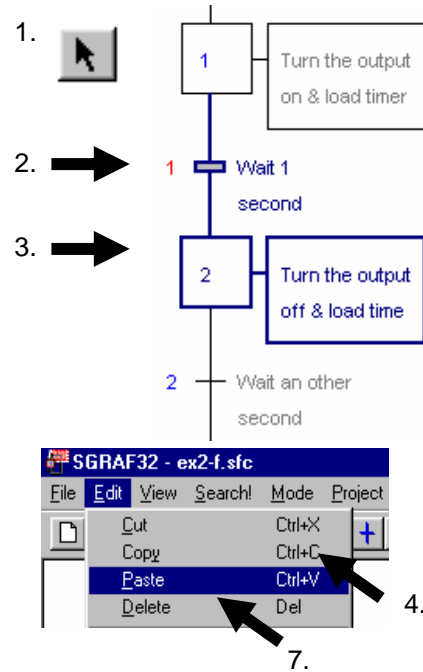
1. Attivare la modalità *Selezione*
2. Fare clic sulla prima transizione della sequenza.
3. Fare clic sull'ultimo passo della sequenza che si desidera cancellare, tenendo premuto il tasto *Shift*.
4. Premere il tasto *Del*.



### 6.7.11 Copiare/Incollare una sequenza

#### Copiare una sequenza:

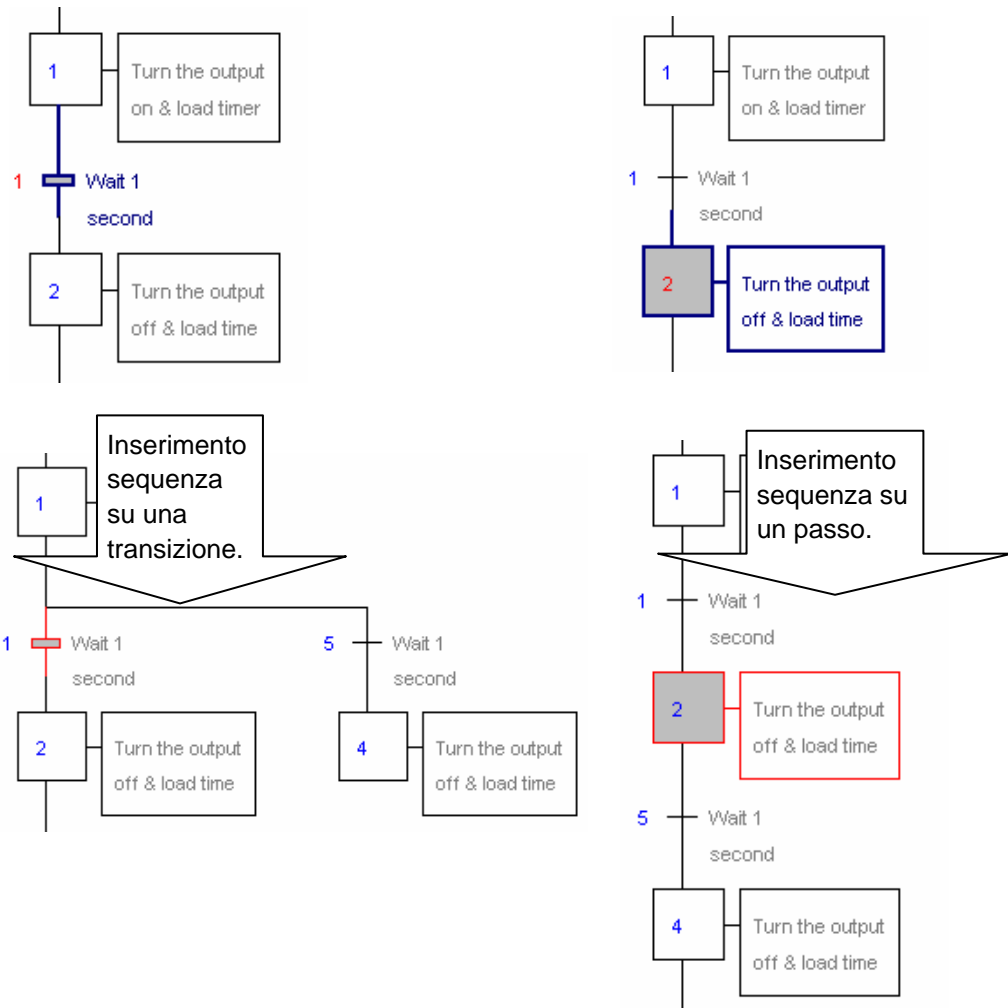
1. Attivare la modalità *Selezione*
2. Contrassegnare l'inizio della sequenza.
3. Fare clic sull'ultimo passo della sequenza tenendo premuto il tasto *Shift*.
4. Selezionare la voce *Copy (Copia)* nel menù *Edit (Modifica)*
5. Attivare la modalità *Selezione*
6. Fare clic sul punto in cui si desidera inserire la sequenza.
7. Selezionare la voce *Paste (Incolla)* nel menù *Edit (Modifica)*.



#### Incollare una sequenza:

##### Nota:

In base alla posizione dell'elemento (transizione o passo) che si desidera inserire, la sequenza può essere inserita sotto o accanto l'elemento selezionato.



## 6.8 Scrittura del primo Blocco Sequenziale

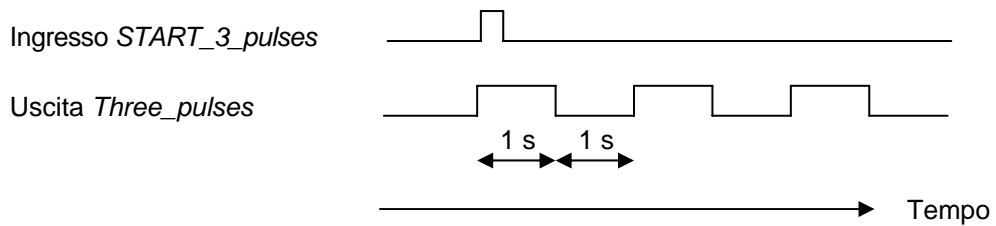
### 6.8.1 Apertura del file

Aprire il file *PULSE.sfc*. Selezionare l'elenco dei blocchi SB e caricare il blocco SB denominato *PULSE*

Obiettivo:

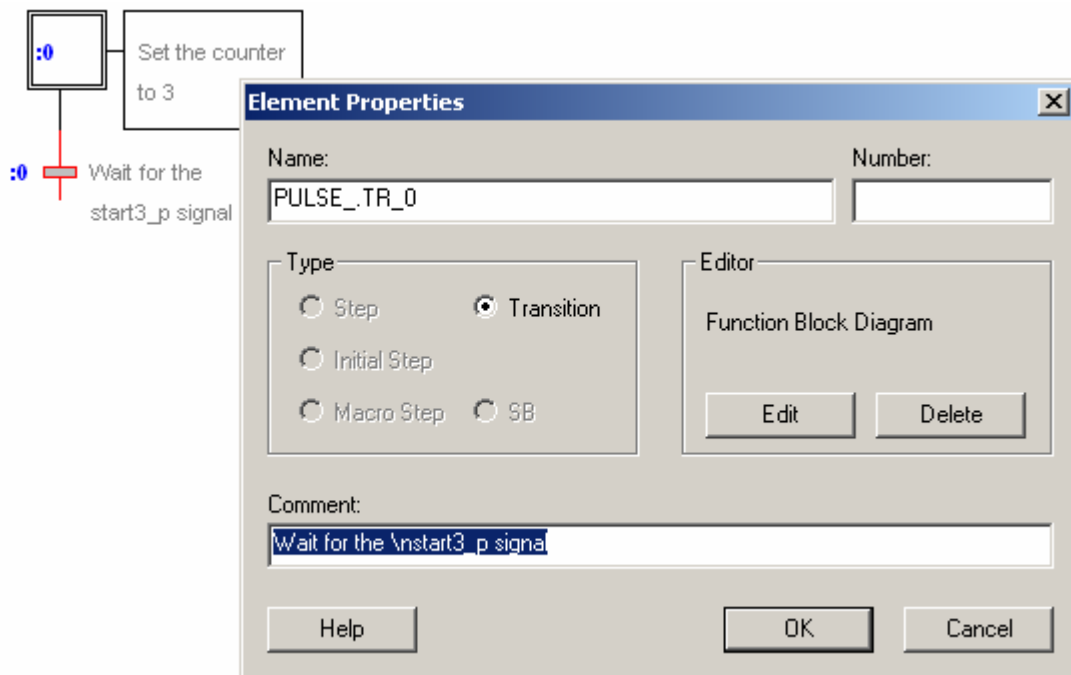
Realizzeremo un programma che faccia lampeggiare per tre volte un'uscita digitale (*Three\_pulses* O 33) ogni volta che l'ingresso digitale (*Start\_3\_pulses* I 2) commuta a livello alto.

Diagramma

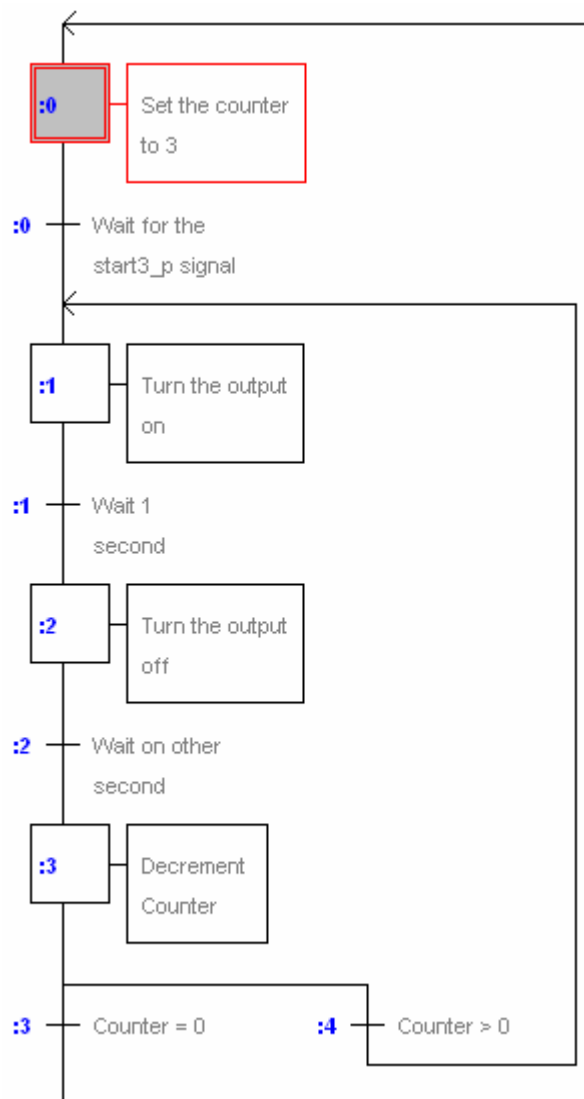


### 6.8.2 Definizione della Struttura Base

La partenza è sempre data da un passo iniziale: il punto di inizio dopo una partenza a freddo. Dopo aver eseguito l'inizializzazione, si può attendere il segnale di partenza: *Start\_3\_pulses*. Inserire gli elementi come illustrato in figura ed aggiungere i commenti ai vari campi:

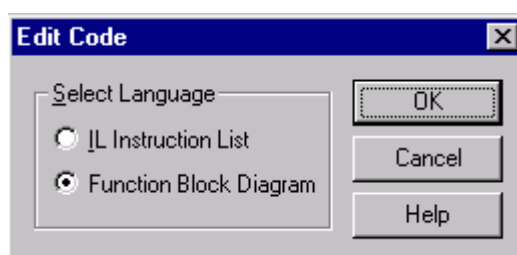


Una volta avviata la sequenza, attiviamo l'uscita *Three\_pulses* per 1 secondo. Quindi la disattiviamo per 1 secondo. Ripetiamo questa operazione per tre volte, quindi riavviamo la sequenza:



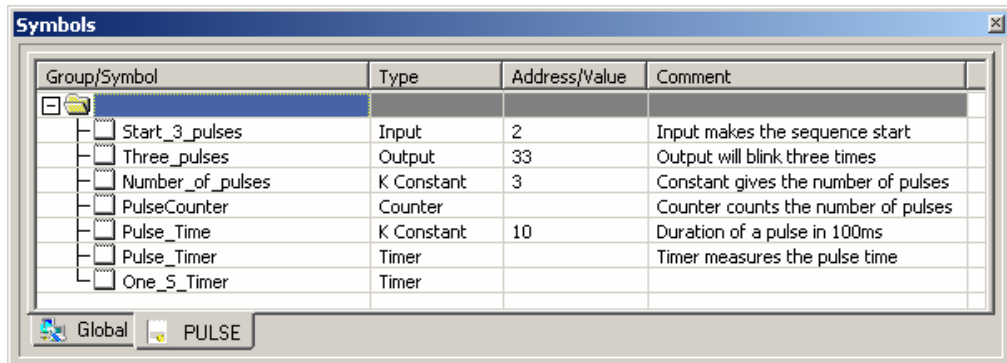
### 6.8.3 Scelta dell'editore: IL o Fupla

A questo punto la sequenza è finita. Non rimarrà che inserire le istruzioni del programma in ogni passo e transizione. Per fare ciò potremo utilizzare indifferentemente l'editore Lista Istruzioni o Fupla. Programmeremo il nostro passo Iniziale utilizzando l'editore Fupla. Fare doppio clic sul Passo Iniziale e scegliere l'editore Fupla:



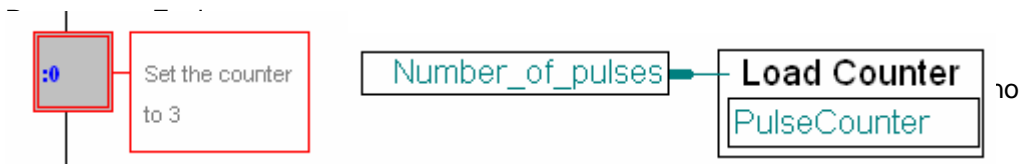
### 6.8.4 Preparazione dei Simboli

Per prima cosa, è necessario stabilire un elenco con tutti gli elementi che intendiamo usare nell'Editore dei Simboli. Inserire gli elementi come illustrato nella figura seguente.

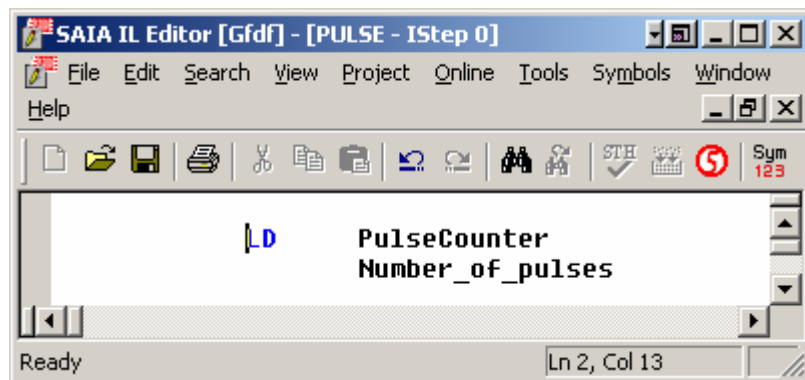


### 6.8.5 Scrittura del programma

Quindi dovremo caricare il contatore denominato *PulseCounter* con la costante 3 denominata *Number\_of\_pulses*.



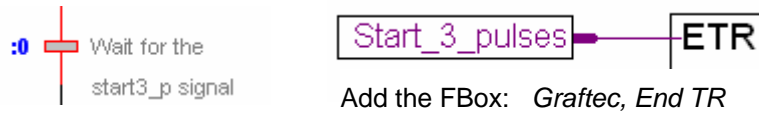
Programma in Lista di Istruzioni:



### 6.8.6 Come programmare una transizione

Una transizione viene ripetuta in continuazione finché non è attiva l'istruzione di fine transizione *ETR* (Programma Fupla) o l'accumulatore (programma in Lista Istruzioni). Nella transizione numero 0 attenderemo all'interno della transizione stessa finché l'ingresso *Start\_on 3\_pulses* non commuta a livello alto. Aprire la transizione 0 ed aggiungere il programma seguente:

*Programma in Fupla:*



*Programma in Lista Istruzioni:*

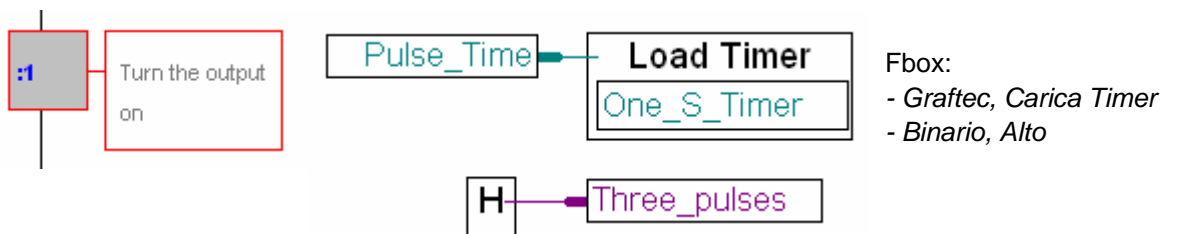
STH Start\_3\_pulses ; Copiare l'informazione Start\_3\_pulses in accu

### 6.8.7 Utilizzo dei Temporizzatori in un SB

Procediamo nel modo seguente: attiviamo l'uscita e carichiamo il temporizzatore in un passo, quindi passiamo alla transizione di attesa, che interroga il temporizzatore fino al termine del ritardo (temporizzatore = 0).

*Programma Fupla:*

I temporizzatori e contatori della Libreria Fupla non sono stati realizzati per essere utilizzati nei blocchi SB. Essi sono destinati ad essere utilizzati in COB eseguiti ciclicamente. Se si vogliono utilizzare dei temporizzatori o contatori all'interno di un blocco SB, si devono utilizzare quelli della famiglia "Graftec". Questi ultimi sono stati ideati appositamente per i Blocchi Sequenziali, in quanto possono essere caricati all'interno di un passo, mentre l'interrogazione sul loro stato può essere effettuata successivamente in un altro passo o transizione.

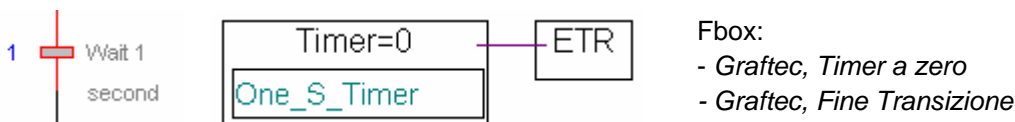


*Programma in Lista Istruzioni:*

SET Three\_pulses ; Imposta l'uscita a livello alto  
LD One\_S\_Timer ; Carica il temporizzatore  
Pulse\_Time

Attesa decremento del temporizzatore:

*Programma Fupla:*



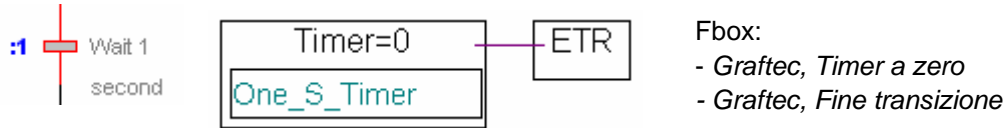
*Programma in Lista Istruzioni:*

STL One\_S\_Timer ; Imposta l'accu a livello alto alla  
; fine del temporizzatore

### 6.8.8 Ripetizione del passo e della transizione per tutto il tempo in cui l'impulso è disattivato

Il Passo 2 e la Transizione 2 sono identici al Passo 1 ed alla Transizione 1.

*Programma Fupla:*

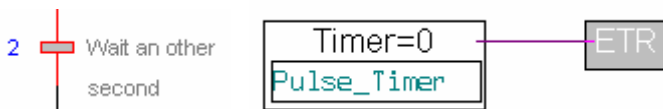


*Programma in Lista Istruzioni:*

```
RES    Three_pulses    ; Imposta l'uscita a livello basso
LD     Pulse_Timer     ; Carica il temporizzatore
      Pulse_Time
```

TR:

*Programma Fupla:*



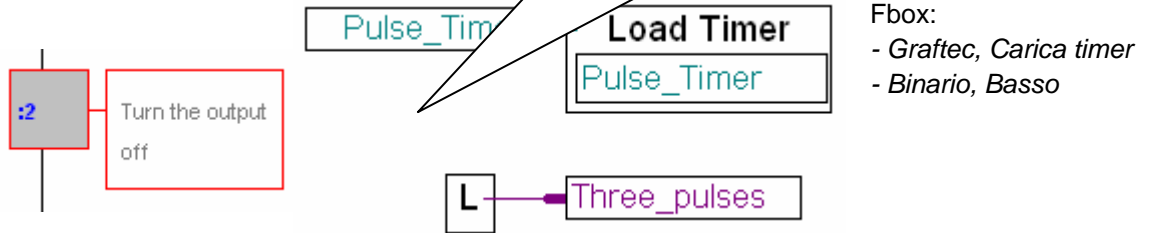
*Programma in Lista Istruzioni:*

```
STL Pulse_Timer          ; Attiva l'accu quando scade il tempo
                        ;
```

**Nota:** Per ST/TR 1 e 2 sono stati usati due temporizzatori diversi (*One\_S\_Timer* e *Pulse\_Timer*). Tuttavia, per risparmiare i temporizzatori, avremmo potuto utilizzare due volte lo stesso temporizzatore (*One\_S\_Timer* o *Pulse\_Timer*), dal momento che gli stessi non sono usati simultaneamente, ma uno dopo l'altro!

### 6.8.9 Decremento di un contatore

Programma Fupla:



Programma in Lista Istruzioni:

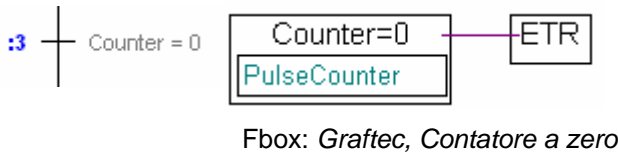
DEC PulseCounter ;Se accu è ancora a 1, decrementare il contatore.

(N.B.: L'accu è sempre a 1 all'inizio di un ST/TR)

### 6.8.10 Diramazione alternativa

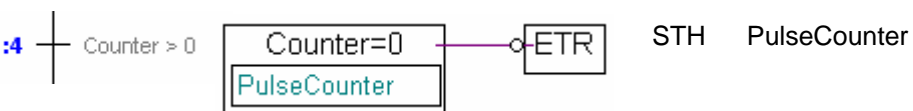
Le ultime due Transizioni sono:

Programma Fupla:



Programma in Lista Istruzioni:

STL PulseCounter



Transizione 3: l'ingresso ETR è attivo se il valore del contatore è zero.

Transizione 4: l'ingresso ETR è attivo se il valore del contatore è diverso da zero.



Inverti  
Connessione  
Binaria

Inserire una inversione all'ingresso della funzione ETR con il pulsante: *Inverti connessione binaria.*



## 6.9 Costruzione (Build) e debug del programma



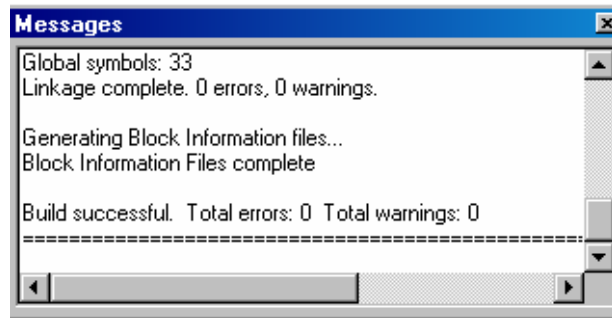
Costruisci  
Tutto

Dopo aver completato la scrittura del programma, sarà necessario compilarlo facendo clic sul pulsante "Build" (Costruisci).

### 6.9.1 Finestra Messaggi

La seguente finestra messaggi fornirà tutte le informazioni necessarie.

Se il programma è stato inserito correttamente, la finestra indicherà:



Build successful. Total errors: 0 Total warnings: 0  
(Costruzione eseguita con successo. N° Totale Errori : 0 : N° Tot. Avvertimenti : 0)

Gli eventuali errori verranno indicati con caratteri rossi. Fare doppio clic sul messaggio di errore per individuare il punto errato nel programma.

### 6.9.2 Strumenti Online



Trasferisci  
Programma

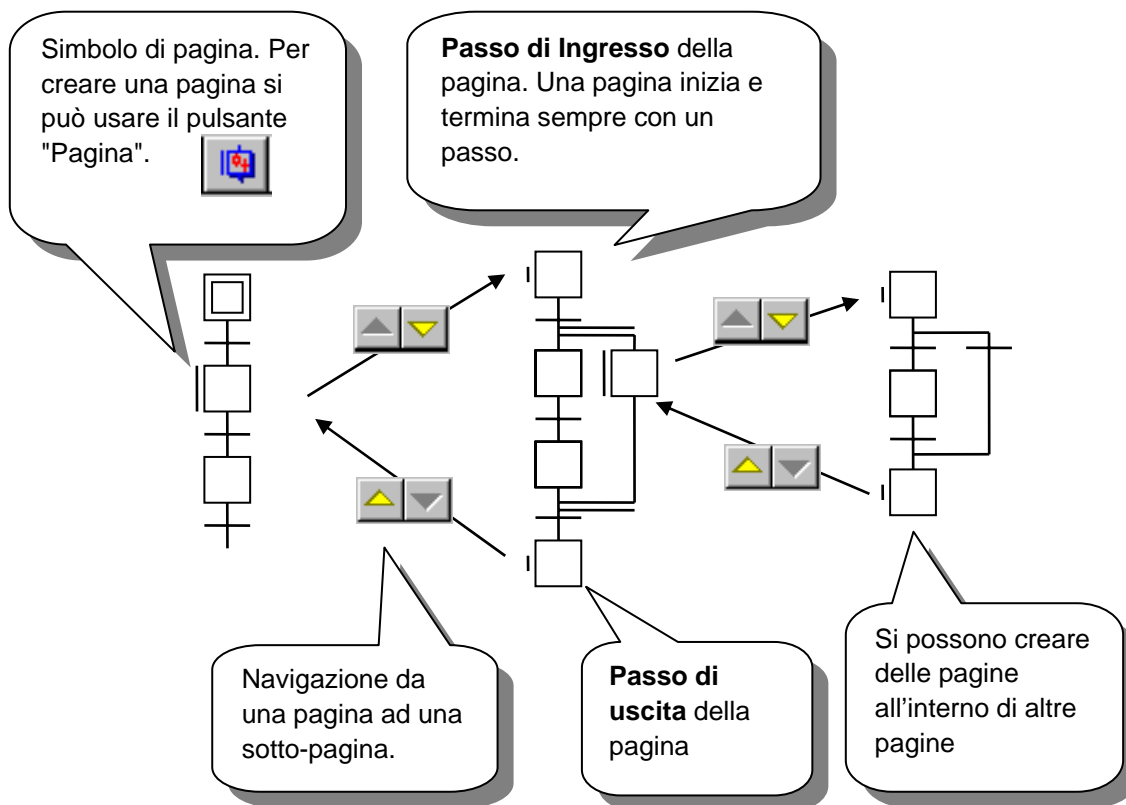
Al termine eseguire il trasferimento del programma ed attivare la modalità Online. Il Blocco Sequenziale può essere visualizzato in modalità Online. Il pallino rosso indica sempre qual è la transizione o il passo attivo.

Il PLC può essere arrestato in qualsiasi momento per proseguire l'esecuzione in modalità passo-passo.

## 6.10 Come strutturare un programma GRAFTEC in pagine

### 6.10.1 Definizione di una pagina

In PG5, i programmi di grosse dimensioni possono essere strutturati in modo da creare una semplice struttura generale che richiami delle sotto-pagine.



Regole per l'uso delle pagine:

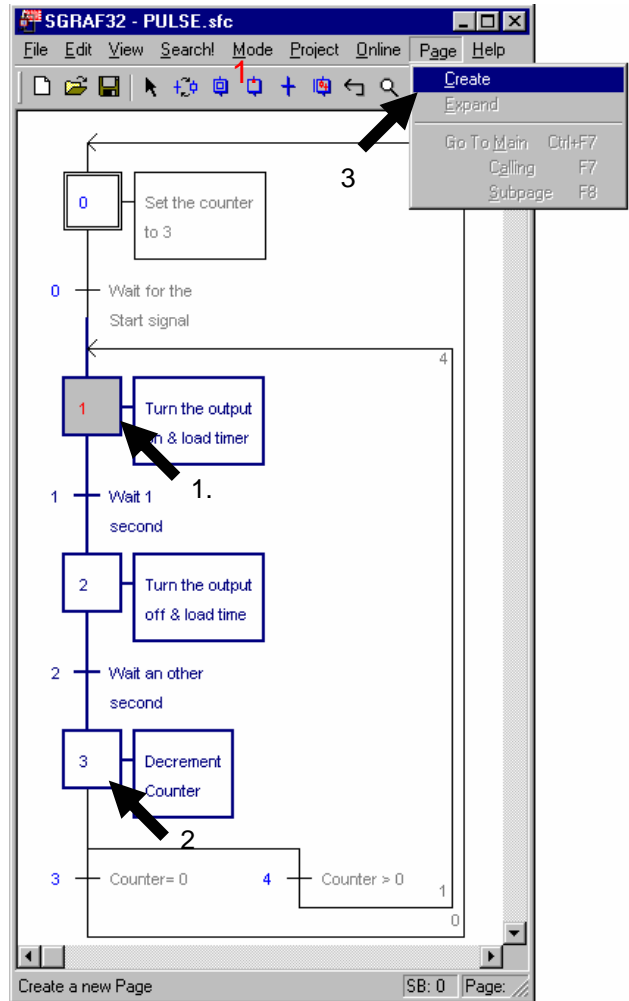
- Una pagina inizia e termina sempre con un Passo
- Una pagina può avere solo un "Passo di Ingresso" e solo un "Passo di Uscita"
- Una pagina può contenere altre pagine (a piacimento)
- Un "passo di Ingresso/Uscita" non può mai essere spostato o cancellato.

### 6.10.2 Modifica di una pagina

*Creazione di una Pagina:*

Creare una pagina da una sequenza.

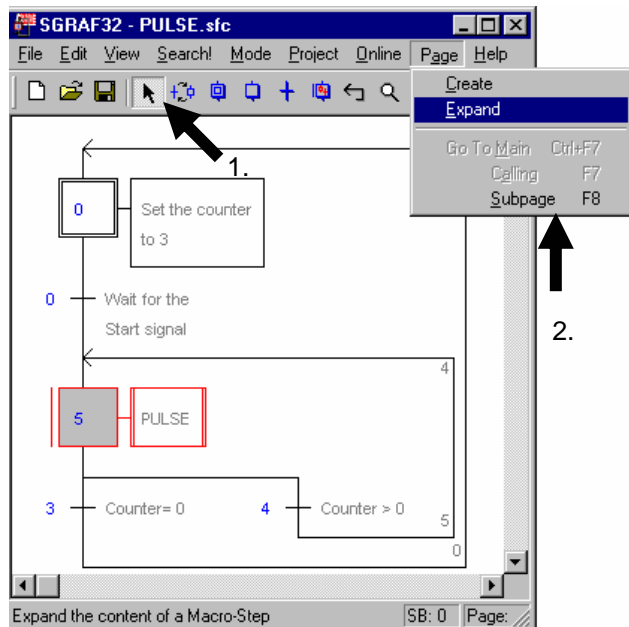
1. Attivare la modalità *Selezione*
2. Fare clic sul primo passo della sequenza
3. Premere il tasto *Shift* e fare clic sull'ultimo passo della sequenza
4. Selezionare la funzione *Create* (*Crea*) nel menù *Page* (*Pagina*).



*Apertura di una pagina:*

Visualizzare il contenuto rappresentato dal simbolo della pagina.

1. Attivare la modalità *Selezione*
2. Fare clic sulla pagina
3. Eseguire la funzione *Subpage* (*Sottopagina*) nel menù *Page* (*Pagina*).



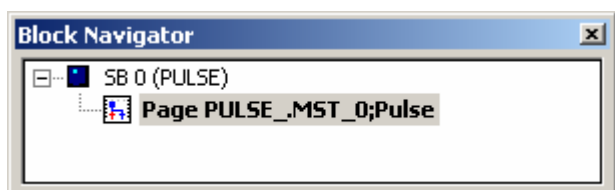
*Annullamento di una Pagina:*

Sostituire la pagina con l'intera sequenza.

1. Attivare la modalità *Selezione*
2. Fare clic sulla pagina
3. Eseguire la funzione *Expand* (*Espandi*) nel menù *Page* (*Pagina*).

**Nota:**

Il *Block Navigator* (*Navigatore Blocchi*) semplifica la navigazione tra le pagine contenenti SB.





## Indice

---

7	Programmazione in Lista Istruzioni (IL)	7-3
7.1	Introduzione	7-3
7.2	Preparazione di un progetto IL	7-4
7.2.1	Creare il nuovo progetto	7-4
7.2.2	Creare un nuovo file IL	7-4
7.3	Organizzazione di una finestra dell'editore IL	7-5
7.3.1	Scrittura di una riga di codice programma	7-6
7.3.2	Formattazione delle righe di istruzioni nella pagina	7-7
7.3.3	Scrittura dei blocchi organizzativi	7-7
7.3.4	Sequenza di elaborazione per istruzioni e blocchi	7-7
7.3.5	Regole da seguire durante la scrittura dei blocchi	7-8
7.4	<i>Finestra Symbols (Simboli)</i>	7-9
7.4.1	Come aggiungere un nuovo Simbolo nella finestra <i>Symbols</i>	7-10
7.4.2	Modalità di indirizzamento degli operandi	7-11
7.4.3	Utilizzo di un Simbolo dell'elenco <i>Symbols</i> in un programma IL	7-12
7.4.4	Simboli locali e globali	7-13
7.5	Introduzione al repertorio di istruzioni PCD	7-14
7.5.1	Accumulatore	7-14
7.5.2	Istruzioni binarie	7-15
7.5.3	Dinamizzazione	7-19
7.5.4	Flag di stato	7-20
7.5.5	Istruzioni di tipo Word per i temporizzatori	7-21
7.5.6	Istruzioni per i contatori	7-23
7.5.7	Istruzioni dipendenti dall'accumulatore	7-24
7.5.8	Istruzioni di tipo word per operazioni aritmetiche su interi	7-25
7.5.9	Istruzioni di tipo word per operazioni aritmetiche in virgola mobile	7-26
7.5.10	Conversione di registri in formato intero e virgola mobile	7-26
7.5.11	Registro indice	7-27
7.5.12	Salti di programma	7-28
7.6	Scrittura del primo programma applicativo	7-30
7.7	Costruzione (Build) del programma	7-32
7.8	Trasferimento (Download) del programma nel PCD	7-33
7.9	Debug di un programma	7-33
7.9.1	Visualizzazione del codice compilato	7-33
7.9.2	Pulsanti On/Offline - Run – Stop – Step by step	7-34
7.9.3	Modalità passo-passo	7-35
7.9.4	Breakpoint (Interruzioni)	7-36
7.9.5	Modifica Online del programma	7-37
7.9.6	Visualizzazione e modifica degli stati dei simboli con la "Finestra di Monitoraggio" ( <i>Watch Window</i> )	7-38
7.10	Messa in servizio di un modulo analogico	7-39
7.10.1	Esempio per i moduli di ingressi analogici PCD2.W340	7-39
7.10.2	Esempio per i moduli di uscite analogiche PCD2.W610	7-40



## **7 Programmazione in Lista Istruzioni (IL)**

---

### **7.1 Introduzione**

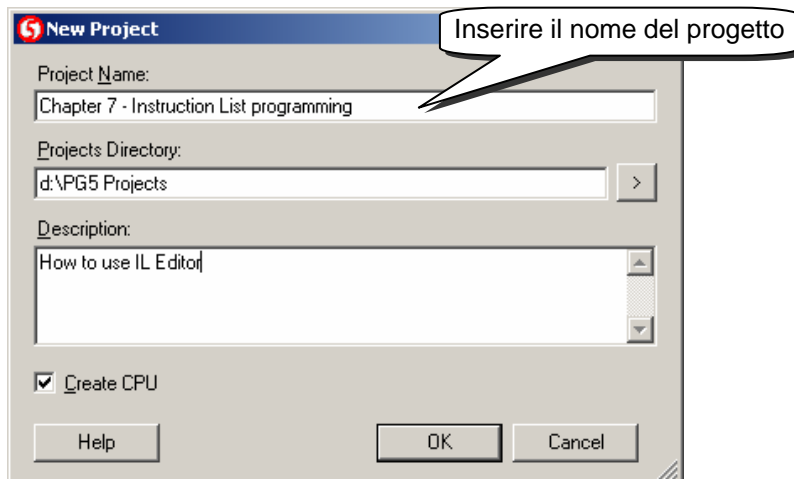
L'Editore per lista istruzioni (IL) è lo strumento più flessibile e potente con cui si possono programmare i controllori PCD. IL sta per "Instruction List" (Lista Istruzioni): un ambiente di programmazione non-grafico che permette all'utente di scrivere i programmi con l'ausilio del potente set di istruzioni dei PCD. Tutti i controllori PCD utilizzano questo set di istruzioni, garantendo quindi la completa portabilità dei programmi da un PCD all'altro. L'Editore per lista istruzioni (IL) non è solo un prezioso ausilio alla stesura dei programmi, ma anche un valido strumento di diagnostica e test on-line.

## 7.2 Preparazione di un progetto IL

Prima di produrre un esempio di programma, si raccomanda di creare un nuovo progetto ed un file in cui editare il programma IL.

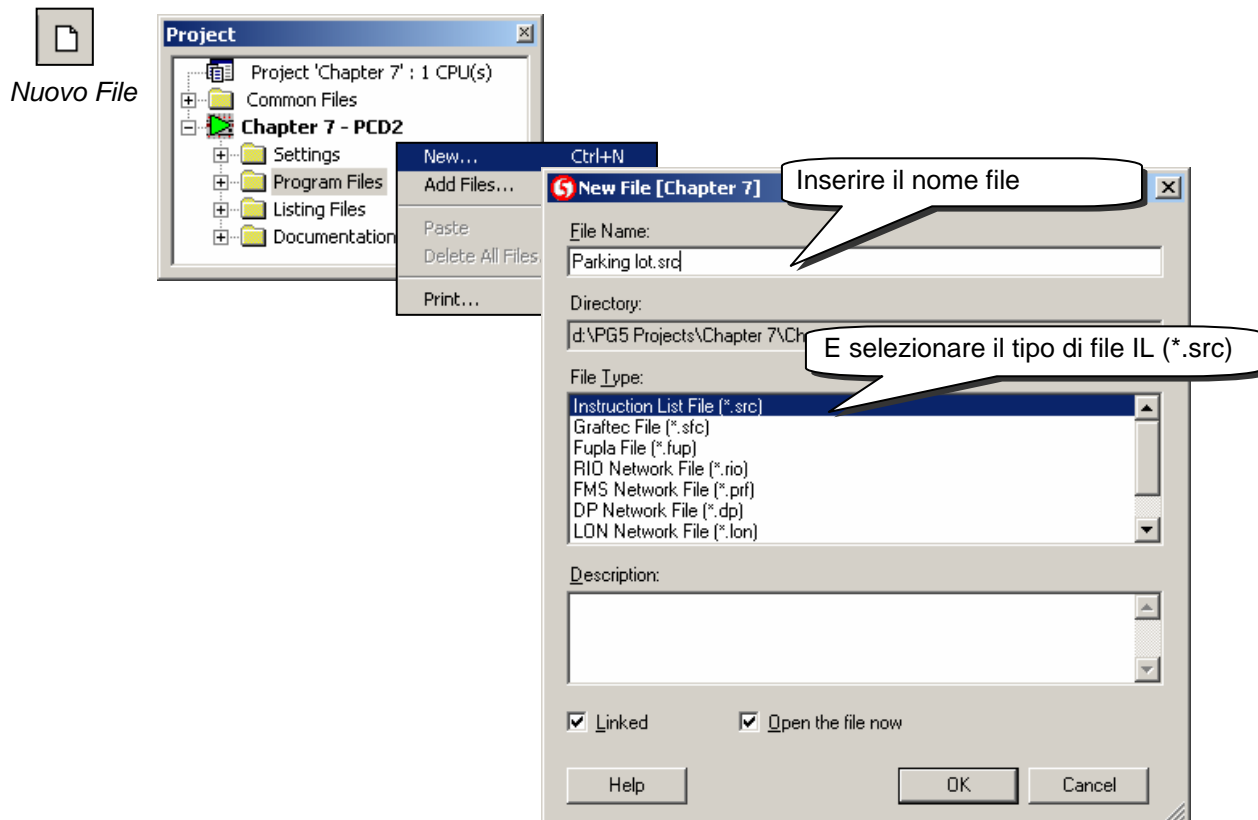
### 7.2.1 Creare il nuovo progetto

Nella finestra SAIA Project Manager (Gestore Progetti SAIA), selezionare la voce di menu Project, New... (Progetto, Nuovo...) per creare il nuovo progetto.



### 7.2.2 Creare un nuovo file IL

Per creare un nuovo file di programma all'interno di questo progetto, selezionare la cartella *Program Files*, fare clic con il pulsante destro del cursore e selezionare la voce di menu *New...* (*Nuovo*) (oppure premere il pulsante *Nuovo File* nella barra degli strumenti):





### 7.3 Organizzazione di una finestra dell'editore IL

**Codice Mnemonico**

Etichette		Operandi	Commenti
<b>Cyclical Organisation Block</b>			
	COB	0 0	; Cyclical program ; No supervision time
	STH	Car_incoming	; A car comes into the parking
	DYN	Dynamise_incoming_car_signal	; On the positiv flank of incor
	DEC	Number_of_free_slots	; Decrement the number of fre
-----			
	STH	Car_outgoing	; A car leaves into the parking
	DYN	Dynamise_leaving_car_signal	; On the positiv flank of outgo
	INC	Number_of_free_slots	; Increment the number of fre
-----			
	STL	Number_of_free_slots	; If no more free parking slots
	OUT	Red_light	; Set the red light
	ECOB		; End of Cyclical program

Inizio COB

Sequenza di elaborazione delle istruzioni del blocco

Fine COB

L'editore IL è simile ad un qualsiasi altro editore di testi disponibile in commercio. Sono infatti presenti le stesse funzioni di elaborazione testi, quali *Taglia/Incolla* o *Trova/Sostituisci*. L'editore IL offre tuttavia molto più di un editore di testi convenzionale:

- Layout di pagina adattato specificamente per la scrittura di programmi PCD
- Colori che facilitano l'identificazione dei vari tipi di informazioni
- Elenco dei simboli utilizzati dal programma, nella finestra Symbols
- Possibilità di visualizzare il programma on-line e di eseguire il test passo-passo

### 7.3.1 Scrittura di una riga di codice programma

Etichetta	Mnemo. Operando	Commento
	<code>;Increment a register</code>	
	<code>STH Flag</code>	<code>;Copy the Flag state into the accu</code>
	<code>DYN DFlag</code>	<code>;On a positiv flank of the Flag  , set the accu eigh</code>
	<code>JR L Next</code>	<code>;If the accu is low, jump to the label Next</code>
	<code>INC Register</code>	<code>; Increment the register</code>
<code>Next:</code>	<code>NOP</code>	<code>;No instruction</code>

Le righe di un programma IL sono formattate in 4 colonne:

Label (Etichetta)

Rappresentata dal colore rosso, l'etichetta è il nome simbolico di una riga di programma. E' utile per le funzioni di salto nell'ambito del programma (JR L Next)

Mnemonic (Codice Mnemonico)

Rappresentato dal colore blu, il codice mnemonico – o l'istruzione di programma – definisce l'operazione da eseguire sull'operando: ingresso, uscita, flag, registro, ...

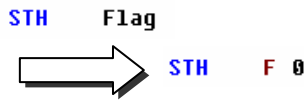
Operand (Operando)

Rappresentato dal colore nero, l'operando definisce il tipo di dato: ingresso, uscita, flag, registro, ... e indirizzo.



Visualizza Simboli o Valori

Il pulsante *Visualizza Simboli o Valori* permette di visualizzare l'indirizzo di un operando o il suo simbolo.



Comment (Commento)

I commenti dell'utente sono visualizzati in verde ed iniziano con un punto e virgola (;). Compaiono alla destra del codice mnemonico e dell'operando, e possono occupare anche l'intera riga.

```

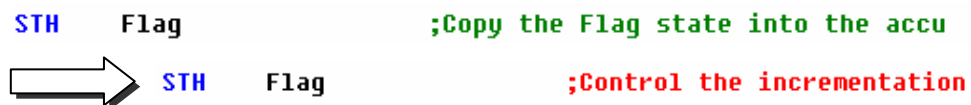
$SKIP
Author: Dupont Fred
Date: 28.10.2003
File: Logic.src
$ENDSKIP
    
```

Se un commento si estende su più righe, non è necessario iniziare sempre ogni riga con il simbolo di punto e virgola. E' possibile infatti inserire il commento tra due istruzioni assembler: \$skip e \$endskip. Queste indicano all'assemblatore di non considerare il testo che compare tra le suddette due istruzioni.



Visualizza Commenti Utente o Automatici

Il pulsante *Visualizza Commenti Utente o Automatici* può essere utilizzato per visualizzare i commenti scritti dall'utente su ogni riga del programma, oppure i commenti automatici, associati ad ogni simbolo utilizzato come operando.

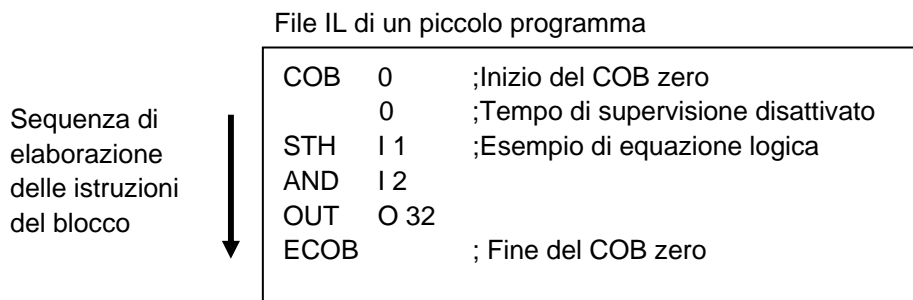


### 7.3.2 Formattazione delle righe di istruzioni nella pagina

Se è selezionata l'opzione *Auto Format while Typing (Formattazione Automatica durante l'Inserimento)*, premendo il tasto *Enter* della tastiera si otterrà la formattazione automatica di ogni riga di programma all'interno della pagina. Vedere la voce di menu *Tools, Options (Strumenti, Opzioni)* nell'editore IL. E' possibile configurare anche la larghezza della colonna.

Se la formattazione della pagina non è appropriata, è possibile contrassegnare con il mouse alcune righe, o tutte le righe del file e riformattarle utilizzando la voce di menu *Tools, Auto Format (Strumenti, Formattazione Automatica)*.

### 7.3.3 Scrittura dei blocchi organizzativi



Il linguaggio di programmazione SAIA PCD è strutturato in blocchi organizzativi, nei quali l'utente scrive i programmi applicativi.

Ogni blocco fornisce un servizio particolare: blocchi ad organizzazione ciclica (COB) per i programmi ciclici; blocchi sequenziali (SB) per i programmi sequenziali; blocchi programma (PB) per le subroutine; blocchi funzione (FB) per le subroutine con parametri; blocchi ad organizzazione eccezionale (XOB) per le routine eccezione.

I blocchi sono delimitati da una istruzione di inizio e da una istruzione di fine. Ad esempio, l'istruzione COB delimita l'inizio di un blocco ad organizzazione ciclica che termina con la stessa istruzione preceduta dalla lettera E per indicare "end" (fine) (ECOB). Tutte le istruzioni del programma che appartengono a questo blocco devono essere poste tra le istruzioni COB e ECOB, quindi non possono mai essere al di fuori del blocco stesso.

Anche il più piccolo programma PCD dovrà avere un COB. Gli altri blocchi possono invece essere aggiunti in base alle esigenze.

### 7.3.4 Sequenza di elaborazione per istruzioni e blocchi

All'interno di ogni blocco, il PCD elabora le istruzioni del programma riga per riga, partendo dalla prima istruzione fino all'istruzione di fine-blocco.

L'ordine con cui vengono scritte le istruzioni all'interno di un blocco organizzativo è importante. Non è invece importante l'ordine con cui vengono scritti i blocchi organizzativi stessi. La sequenza di elaborazione dei blocchi è infatti definita da varie regole:

Durante una partenza a freddo del PCD, il controllore programmabile cerca per prima cosa il blocco XOB 16, cioè il blocco che provvede alla gestione della partenza a freddo. Se questo è presente, verrà sempre eseguito come primo blocco, indipendentemente dal fatto che si trovi all'inizio o alla fine del file.

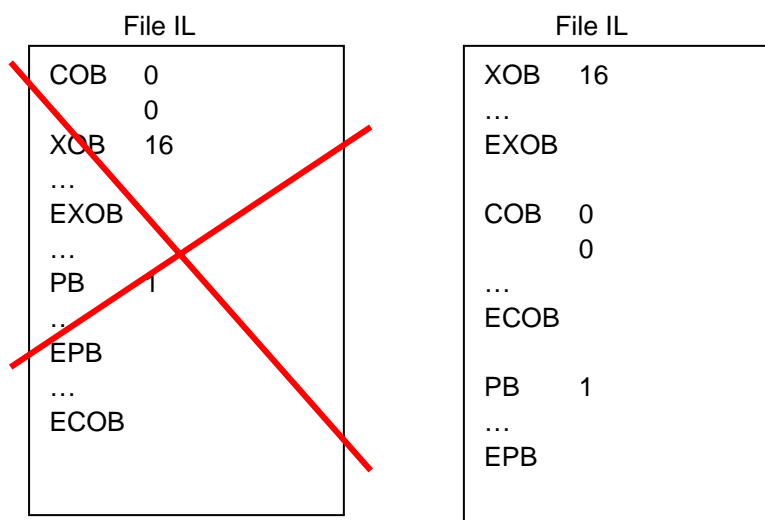
Successivamente, la macchina cerca i vari COB all'interno del programma e li elabora in ordine numerico: COB 0, COB 1, ... COB 15, indipendentemente dall'ordine con cui gli stessi compaiono nel file. Dopo l'esecuzione dell'ultimo COB, il programma ripartirà dal COB 0.

Tutti i blocchi dei programmi sequenziali (SB), subroutine (PB) e funzioni (FB) sono richiamati dal programma utente con le istruzioni CSB (Chiamata SB), CPB (Chiamata PB) e CFB (Chiamata FB). E' il programma utente quindi, che determina quando e in quale ordine devono essere elaborati i blocchi SB, PB e FB.

Tutti i blocchi eccezione vengono automaticamente richiamati non appena si verifica il particolare evento interessato. Questi eventi sono imprevedibili e si possono verificare in qualsiasi momento. L'ordine con cui essi vengono elaborati non può essere definito a priori. Ogni evento hardware o software è legato ad un XOB distinto. Questi eventi non possono essere modificati dall'utente. Tuttavia, l'utente è libero di programmare quale azione deve essere intrapresa all'interno di ogni XOB.

### 7.3.5 Regole da seguire durante la scrittura dei blocchi

Anche se i blocchi possono essere scritti in qualsiasi ordine, devono sempre essere rispettate le seguenti regole:



Non è possibile scrivere dei blocchi all'interno di altri blocchi. I blocchi devono quindi essere sempre consecutivi.

Nessuna istruzione di programma può essere definita all'esterno di un blocco, ad eccezione delle definizioni dei simboli, dei testi e dei data block.

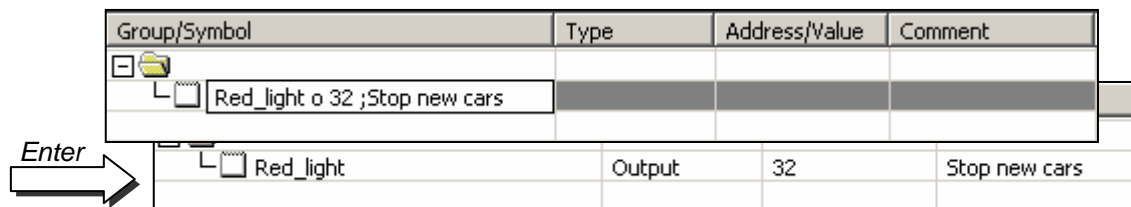


### 7.4.1 Come aggiungere un nuovo Simbolo nella finestra *Symbols*

Metodo semplice

Per aggiungere un simbolo all'elenco aprire la finestra *Symbols*, posizionare il mouse al centro della finestra e fare click con il pulsante destro per selezionare il menu di contesto *Insert Symbol (Inserisci Simbolo)*. A questo punto compilare i campi: *Group/Symbol*, *Type*, *Address/Value* e *Comment* (*Gruppo/Simbolo*, *Tipo*, *Indirizzo/Valore* e *Commento*).

Metodo rapido 1



Group/Symbol	Type	Address/Value	Comment
Red_light o 32 ;Stop new cars			
Red_light	Output	32	Stop new cars

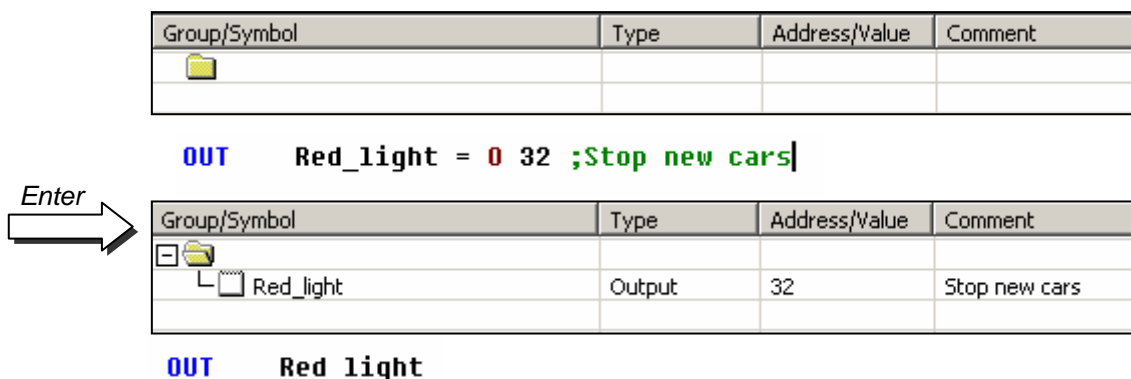
E' possibile anche inserire delle variabili per i vari campi di informazione a partire dal campo *Group/Symbol*. Questo è un metodo più pratico e veloce. Vedere l'esempio seguente.

Sintassi da seguire:

nome\_simbolico tipo indirizzo ;commento

Se il nuovo simbolo è stato definito utilizzando la sintassi sopra indicata, le informazioni verranno inserite nei campi corretti semplicemente premendo il tasto *enter* sulla tastiera.

Metodo rapido 2



Group/Symbol	Type	Address/Value	Comment
OUT Red_light = 0 32 ;Stop new cars			
Red_light	Output	32	Stop new cars

Durante la fase di editing del programma si possono anche aggiungere nuovi simboli. Per fare questo, scrivere una riga di codice programma contenente il codice mnemonico e per l'operando, inserire il nome simbolico e la definizione utilizzando la seguente sintassi:



nome\_simbolico tipo indirizzo ;commento

Il nuovo simbolo verrà automaticamente inserito nell'elenco della finestra *Symbols*, premendo il tasto *enter* sulla tastiera; in questo caso solo se la definizione del simbolo è corretta, e solo se è stata selezionata l'opzione *Automatically add entered type/value to the Symbol Table (Aggiunta automatica del tipo/valore inserito nella Tabella Simboli)* (menu *Tools, Options - Strumenti, Opzioni* nell'editor IL).

## 7.4.2 Modalità di indirizzamento degli operandi



La definizione di un simbolo non include necessariamente tutte le informazioni sotto indicate. Possiamo distinguere fra tre tipi di indirizzamento:

Indirizzi assoluti

Group/Symbol	Type	Address/Value	Comment
 	Output	32	Stops new cars



I dati sono definiti solo con tipo e indirizzo (es. 32), ed un commento opzionale. L'utilizzo dell'indirizzamento assoluto direttamente nel programma si traduce in uno svantaggio quando si modifica il tipo o l'indirizzo. Il programma utente non verrà aggiornato dalle modifiche effettuate nell'elenco dei simboli. Le modifiche devono essere effettuate manualmente per ogni riga del programma. E' quindi preferibile utilizzare nomi simbolici, con indirizzamento dinamico opzionale.

Nomi simbolici

Group/Symbol	Type	Address/Value	Comment
  red_light	Output	32	Stops new cars

I dati sono definiti con un nome simbolico, tipo, indirizzo e commento opzionale. La correzione di un simbolo, tipo o indirizzo è supportata dall'elenco dei simboli ed ogni riga del programma utente viene automaticamente aggiornata al variare del simbolo.

Indirizzamento dinamico

Group/Symbol	Type	Address/Value	Comment
  red_light	F		Stops new cars

Si tratta di una forma di indirizzamento simbolico in cui l'indirizzo non è definito. L'indirizzo viene assegnato automaticamente durante la costruzione del programma. L'indirizzo viene acquisito in un campo di indirizzi definiti in *Software Settings (Impostazioni Software)*. (Vedere Project Manager.)

N.B.: L'indirizzamento dinamico è disponibile per flag, contatori, temporizzatori, registri, testi, DB, COB, PB, FB e SB. Per ingressi, uscite e XOB è necessario invece definire sempre indirizzi assoluti.

### 7.4.3 Utilizzo di un Simbolo dell'elenco *Symbols* in un programma IL

Durante la scrittura di un programma, possono essere utilizzati in vari modi i simboli già definiti nella finestra *Symbols*:

Inserimento di un simbolo da tastiera

Il nome del simbolo viene inserito interamente da tastiera per ogni istruzione che ne fa uso. Questo metodo potrebbe comportare degli errori di digitazione durante la fase di scrittura del nome simbolico, e questo verrebbe evidenziato solo all'atto della costruzione del programma.

Inserimento di un simbolo tramite ricerca selettiva

The diagram illustrates the process of selecting a symbol from a list. It shows a table with columns: Group/Symbol, Type, Address..., and Comment. The table lists symbols like Car\_incoming, Car\_outgoing, Red\_light, Number\_of\_free\_slots, Dynamise\_incoming\_car\_signal, and Dynamise\_leaving\_car\_signal. A 'Choose a Symbol' dialog box is shown with a search filter 'sth Dyn'. The results in the dialog are 'Dynamise\_incoming\_car\_signal F ;Flag' and 'Dynamise\_leaving\_car\_signal F ;Flag'. A callout box shows 'Ctrl + Spazio' and 'sth Dyn' leading to the dialog. Another callout shows '↑, ↓, Enter' leading to the selected symbol 'DYN Dynamise\_leaving\_car\_signal'.

Group/Symbol	Type	Address...	Comment
Car_incoming	Input	0	Gets high when a
Car_outgoing	Input	1	Gets high when a
Red_light	Output	32	Stops new cars a
Number_of_free_slots	Counter		Counts the numb
Dynamise_incoming_car_signal	F		Flag detects the
Dynamise_leaving_car_signal	F		Flag detects the

Choose a Symbol

sth Dyn

↑, ↓, Enter

DYN Dynamise\_leaving\_car\_signal

Se, dopo aver digitato solo alcuni caratteri del nome simbolico da tastiera si premono i tasti *Ctrl+Barra spaziatrice*, compare una finestra contenente l'elenco di tutti i simboli che iniziano con le lettere digitate. Il simbolo richiesto può quindi essere selezionato con il mouse o con i tasti freccia della tastiera (↑, ↓) e confermato premendo *Enter*.

Inserimento di un simbolo con il metodo "trascina e rilascia"

The diagram shows the 'drag and drop' method. A callout box says 'Posizionare il cursore del mouse sul simbolo, premere e tenere premuto il pulsante sinistro'. Below it, a table shows 'DynamiseCar\_Leaving' selected. A callout box says 'Trascinare il cursore del mouse nell'editor IL'. An arrow points to the result 'sth DynamiseCar\_Leaving'. A final callout box says 'Rilasciare il pulsante del mouse'.

Group	Type	Address/...	Comment
DynamiseCar_Leaving	F		Flag detect...

sth

Trascinare il cursore del mouse nell'editor IL

Rilasciare il pulsante del mouse

sth DynamiseCar\_Leaving

Questo metodo di utilizzo del simbolo esclude qualsiasi possibilità di errori di battitura. Nella finestra *Symbols*, posizionare il cursore del mouse sulla riga di definizione di un simbolo, premere e tenere premuto il pulsante sinistro del mouse. Trascinare il cursore del mouse all'interno dell'editor IL e rilasciare il pulsante del mouse. Il simbolo scelto verrà automaticamente aggiunto nella posizione indicata dal cursore del mouse.



### 7.4.4 Simboli locali e globali

La finestra di definizione simboli contiene due cartelle: *Globale* e *Locale*



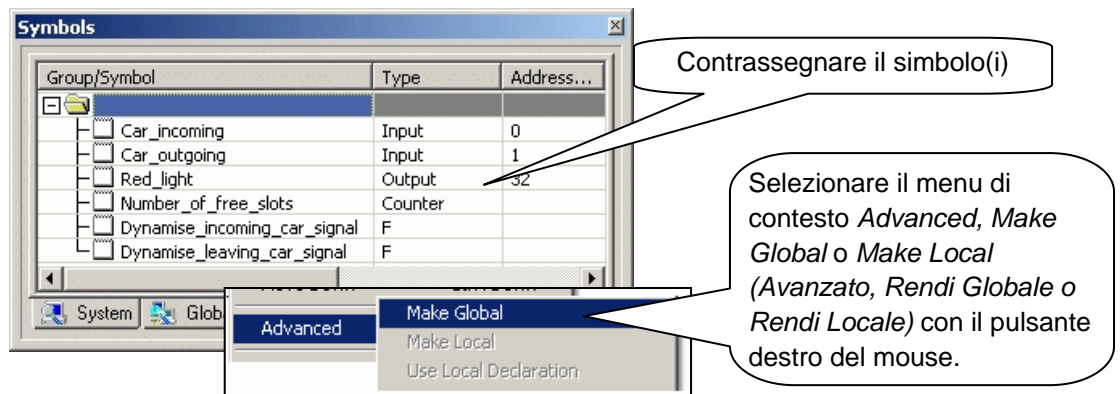
#### Definizione

I simboli locali compaiono in una cartella contenente il nome del file che li utilizza. Questi simboli possono essere utilizzati esclusivamente all'interno di quel file. (*Parking lot.src*)

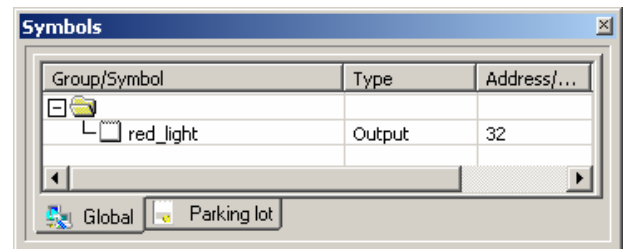
I simboli Globali che compaiono nella cartella *Globale* possono essere utilizzati da tutti i file all'interno della CPU (*Parking lot.src* e *Ventilation.src*)

#### Conversione Locale/Globale

Se necessario, i simboli della finestra *Symbols* possono essere spostati dalla cartella locale alla cartella globale e viceversa.



Il simbolo viene spostato nella cartella *Globale* o *Locale*



Qualsiasi nuovo simbolo definito direttamente dall'editore IL verrà aggiunto nella cartella Globale o nella cartella Locale, in base alle impostazioni effettuate nell'opzione *Global symbols (Simboli Globali)*. Vedere il menu di contesto *Advanced, Options (Avanzate, Opzioni)* nella finestra *Symbols*.

## 7.5 Introduzione al repertorio di istruzioni PCD

Questa sezione fornisce una panoramica generale del repertorio di istruzioni PCD. Per informazioni più dettagliate, consultare la descrizione completa delle singole istruzioni fornita nel *Manuale delle Istruzioni 26/733* oppure nelle schermate di aiuto di PG5. Per ottenere l'aiuto specifico su una istruzione dall'editore IL, operare come segue: scrivere l'istruzione, posizionare il cursore su quest'ultima e premere il tasto *F1*. E' disponibile inoltre l'aiuto generale tramite la voce di menu *Help, Instruction List Help (Aiuto, Elenco Istruzioni)*.

### 7.5.1 Accumulatore

L'accumulatore è un valore binario impostato dalle istruzioni binarie e da alcune istruzioni integer. Il PCD ha un solo accumulatore, che può essere considerato come una flag di tipo speciale. Lo stato dell'accumulatore può essere forzato con l'istruzione *ACC*. L'istruzione *ACC* permette inoltre di forzare l'accumulatore con il valore di una flag di stato (vedere la descrizione delle flag di stato).

**Esempi:**

**ACC H**

Forza l'accumulatore allo stato alto

**ACC L**

Forza l'accumulatore allo stato basso

**ACC C**

Inverte (complementa lo stato dell'accumulatore)

### 7.5.2 Istruzioni binarie

Le istruzioni binarie utilizzano degli operandi che possono avere solo uno di due stati distinti: 0 o 1 (basso o alto). Queste istruzioni sono usate per eseguire equazioni binarie con gli stati di ingressi, uscite, flag, contatori e temporizzatori del PCD.

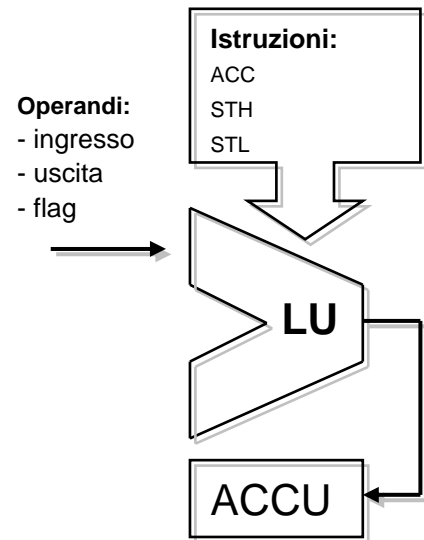
Le istruzioni binarie coinvolgono sempre l'accumulatore. Alcune istruzioni binarie influenzano lo stato dell'accumulatore:

**Esempi:**

**ACC H**  
Forza l'accumulatore allo stato alto

**ACC L**  
Forza l'accumulatore allo stato basso

**STH I 4**  
Copia lo stato dell'ingresso 4 nell'accumulatore. Lo stato dell'accumulatore sarà alto se l'ingresso 4 è a 24 volt. Lo stato dell'accumulatore sarà basso se l'ingresso 4 è a 0 volt.



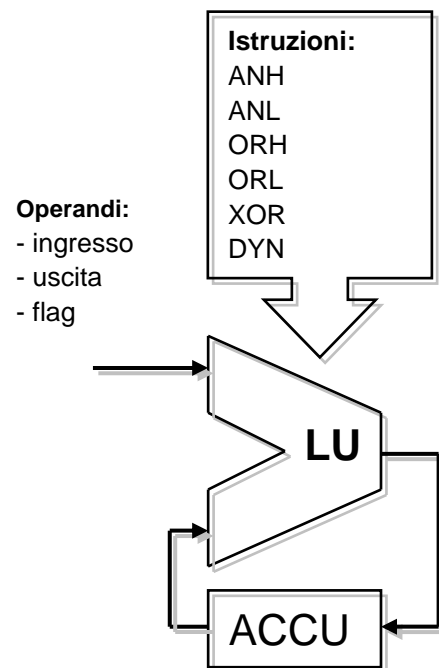
Altre istruzioni leggono lo stato dell'accumulatore per eseguire una funzione binaria e restituiscono il risultato nell'accumulatore:

**Esempi:**

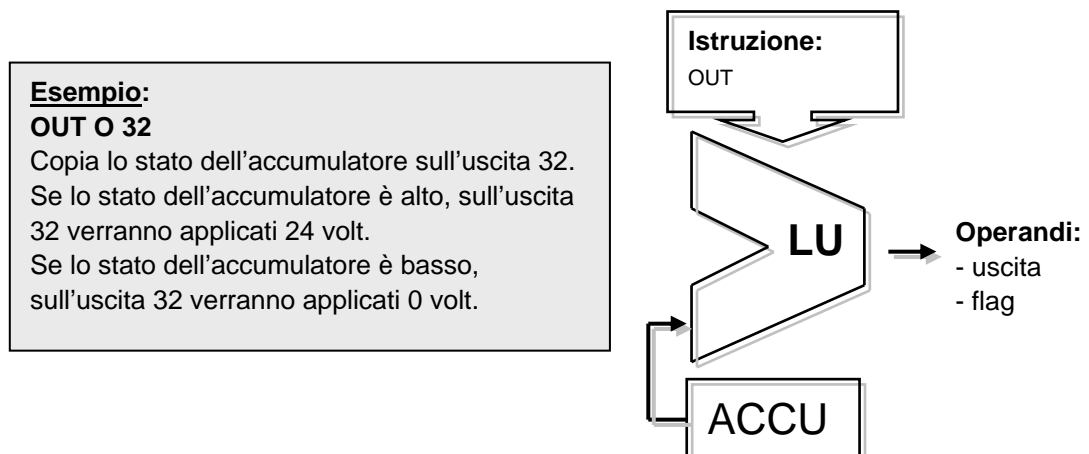
**ANH I 5**  
Legge lo stato dell'accumulatore ed esegue la funzione logica AND con lo stato dell'ingresso 5. L'accumulatore viene impostato con il risultato della funzione.

**ORH F 100**  
Legge lo stato dell'accumulatore ed esegue la funzione logica OR con lo stato della flag 100. L'accumulatore viene impostato con il risultato della funzione.

**XOR T 3**  
Legge lo stato dell'accumulatore ed esegue la funzione logica XOR con lo stato del temporizzatore 3. L'accumulatore viene impostato con il risultato della funzione.

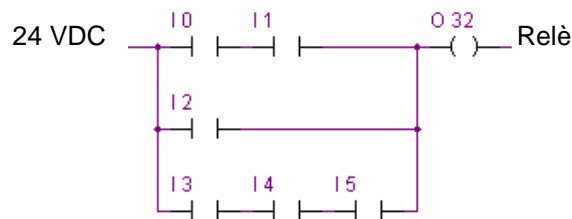


Il risultato di qualsiasi equazione binaria è sempre salvato nell'accumulatore. L'istruzione *OUT* permette di copiare il contenuto dell'accumulatore su una uscita o su una flag:



**Esempio:** programmazione di una semplice equazione binaria.

Questo esempio di programma esegue l'equazione binaria:  $O32 = I0 * I1 + I2 + I3 * I4 * I5$   
 Esso può anche essere rappresentato dallo schema seguente:



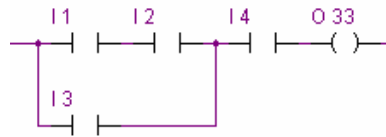
Una equazione binaria inizia sempre con una istruzione *STH* o *STL*, a cui faranno seguito le funzioni *ANH* (\*), *ORH* (+), *XOR* necessarie.

Notare che l'istruzione *ORH* ha la priorità sull'istruzione *ANH*. Ogni istruzione *ORH* contrassegna l'inizio di una nuova linea di contatti nello schema sopra indicato. Il risultato parziale o finale di una equazione binaria viene sempre posto nell'accumulatore. L'istruzione *OUT* permette di utilizzare il risultato dell'accumulatore per modificare lo stato di una uscita o flag.

COB	0	;Inizio di un programma ciclico
	0	
STH	I 0	;Copia lo stato dell'ingresso I 0 nell'accumulatore: Accu = I0
ANH	I 1	;Funzione AND tra lo stato dell'accumulatore e l'ingresso 1:Accu = I0*I1
ORH	I 2	;Funzione OR tra lo stato dell'accumulatore e l'ingresso 2:Accu= I0*I1+I2
ORH	I 3	; Accu = I0*I1+I2+I3
ANH	I 4	; Accu = I0*I1+I2+I3*I4
ANH	I 5	; Accu = I0*I1+I2+I3*I4*I5
OUT	O 32	;Copia il risultato dell'equazione presente nell'accu sull'uscita 32
ECOB		;Fine del programma ciclico

**Esempio:** programmazione di una equazione binaria con ordine di valutazione modificato

Questo esempio di programma esegue l'equazione binaria:  $O33 = (I1 * I2 + I4) * I3$   
 Esso può anche essere rappresentato dallo schema seguente:



Talvolta può essere necessario modificare l'ordine di priorità delle funzioni binarie. Generalmente, questo si ottiene inserendo delle parentesi nelle equazioni. Tuttavia, il repertorio di istruzioni PCD non include le parentesi. L'equazione deve quindi essere suddivisa in due equazioni più piccole. La prima equazione produce il risultato della parte racchiusa tra parentesi e lo salva temporaneamente in una flag, mentre la seconda equazione prende il risultato provvisorio salvato sulla flag e calcola il risultato finale.

```

COB  0
      0
STH  I 1          ;Prima equazione
ANH  I 2
ORH  I 4
OUT  F 0          ;Risultato della funzione tra parentesi: F0 =(I1*I2+I4)

STH  F 0          ;Seconda equazione
ANH  I 3
OUT  O 33         ;Risultato finale: O 33 = F0*I3
ECOB
    
```

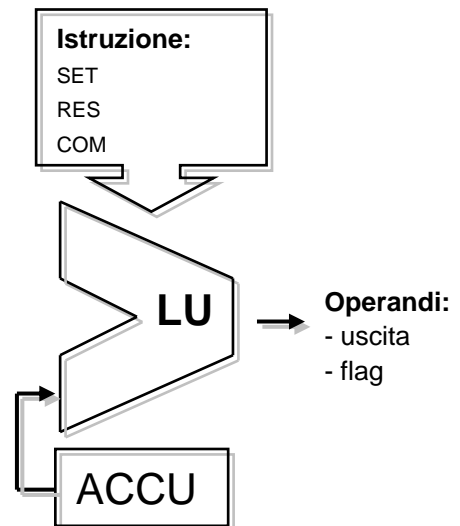
Anche altre istruzioni binarie permettono di utilizzare l'accumulatore per modificare lo stato di una uscita o flag. Ogni istruzione supporta una funzione diversa.

**Esempio:**

**SET O 32**  
 Se lo stato dell'accumulatore è alto, l'uscita 32 verrà forzata a livello alto. In caso contrario l'uscita rimarrà nel suo stato attuale.

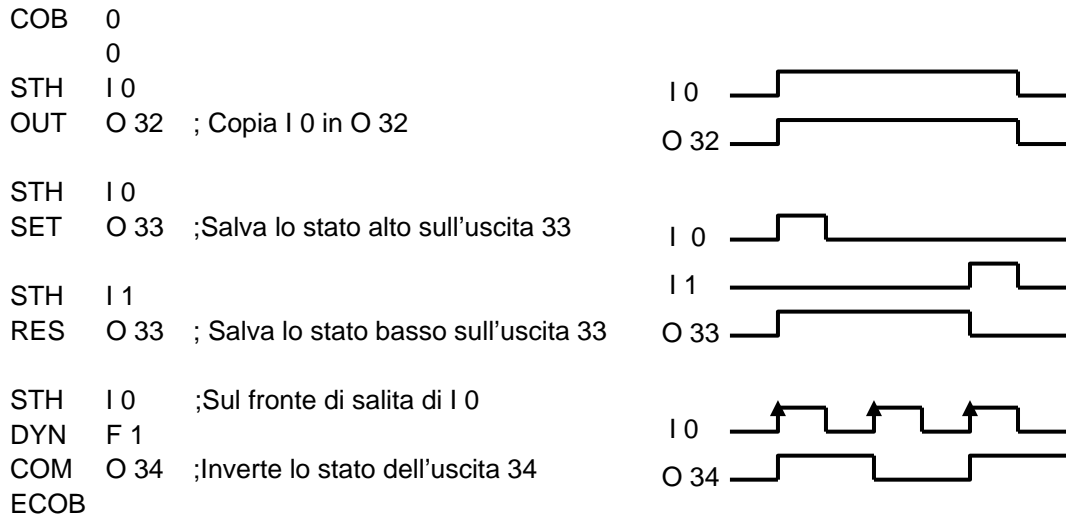
**RES O 32**  
 Se lo stato dell'accumulatore è alto, l'uscita 32 verrà forzata a livello basso. In caso contrario l'uscita rimarrà nel suo stato attuale.

**COM O 33**  
 Se lo stato dell'accumulatore è alto, l'uscita 33 verrà invertita. In caso contrario l'uscita rimarrà nel suo stato attuale.



Esempio:

Questo esempio indica le differenze tra le istruzioni OUT, SET, RES, e COM



Alcune istruzioni binarie terminano con la lettera H o L. Le istruzioni che terminano con L invertono lo stato di qualsiasi informazione prima di eseguire la propria funzione.

**Esempi:**

**STH I 4**

Copia lo stato dell'ingresso 4 nell'accum. Lo stato dell'accumulatore è alto se all'ingresso 4 sono applicati 24 volt.

**STL I 4**

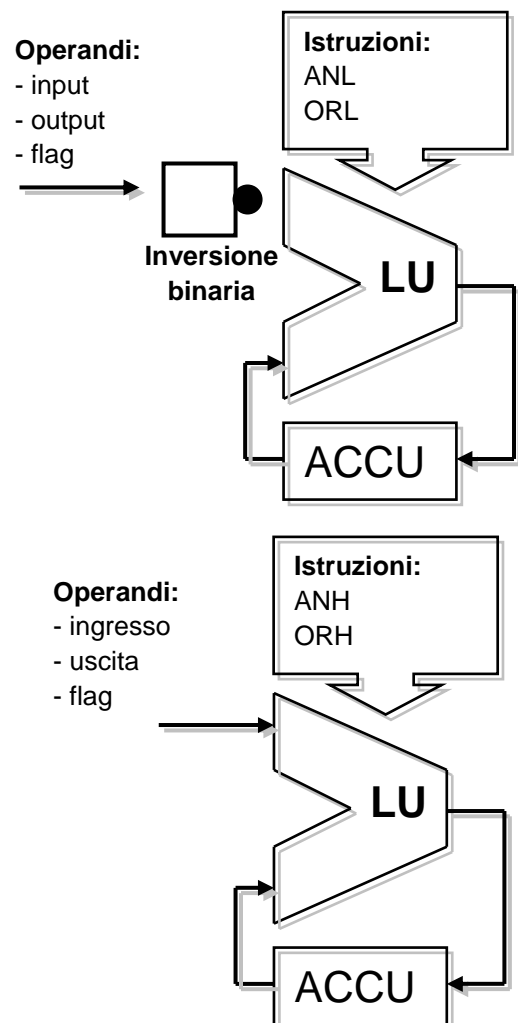
Copia lo stato invertito dell'ingresso 4 nell'accumulatore. Lo stato dell'accumulatore è basso se all'ingresso 4 sono applicati 24 volt

**ANH I 5**

Esegue una funzione logica AND tra lo stato dell'accumulatore e lo stato dell'ingresso 5.

**ANL I 5**

Esegue una funzione logica AND tra lo stato dell'accumulatore e lo stato invertito dell'ingresso 5.



### 7.5.3 Dinamizzazione

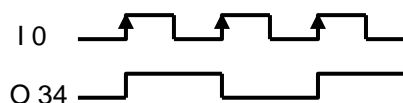
Le istruzioni binarie usano generalmente lo stato binario basso o alto per eseguire una funzione binaria o modificare lo stato di una uscita o di una flag.

Talvolta non è lo stato binario basso o alto che ci interessa, bensì il passaggio da uno stato basso a uno stato alto (ad es. per incrementare un contatore).

Per rilevare un fronte di salita, procedere nel modo seguente: caricare il risultato di una equazione binaria nell'accumulatore ed utilizzare l'istruzione *DYN* per individuare il fronte positivo. Dopo l'istruzione *DYN*, lo stato dell'accumulatore sarà alto se è stato individuato un fronte positivo, in caso contrario sarà basso. La flag utilizzata dell'istruzione *DYN* può essere utilizzata solo per una singola istruzione di dinamizzazione. Questo è dovuto al fatto che viene utilizzata per conservare lo stato per il ciclo di programma successivo.

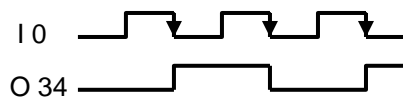
**Esempio:** rilevamento di un fronte di salita

STH I 0  
 DYN F 3  
 COM O 34



**Esempio:** rilevamento di un fronte di discesa

STL I 0  
 DYN F 3  
 COM O 34



Per aiutarvi a verificare l'influenza delle istruzioni *DYN* sul programma sopra illustrato, vi suggeriamo di rimuovere l'istruzione *DYN* ed osservare il comportamento del programma.

### 7.5.4 Flag di stato

Contrariamente alle istruzioni binarie, le istruzioni integer di tipo "word" usano raramente l'accumulatore. Tuttavia, esse modificano quasi sempre le flag di stato.

Le 4 flag di stato del PCD vengono modificate dalle istruzioni "word" e forniscono l'informazione sul risultato.

Flag positivo	P	Impostata se il risultato è positivo.
Flag negativo	N	Impostata se il risultato è negativo
Flag di zero	Z	Impostata se il risultato è zero
Flag di errore	E	Impostata in caso di errore

La flag di errore può essere impostata per svariati motivi, provocando una chiamata al blocco eccezione XOB 13:

Overflow causato da un'istruzione che moltiplica due numeri di valore elevato

Divisione per zero

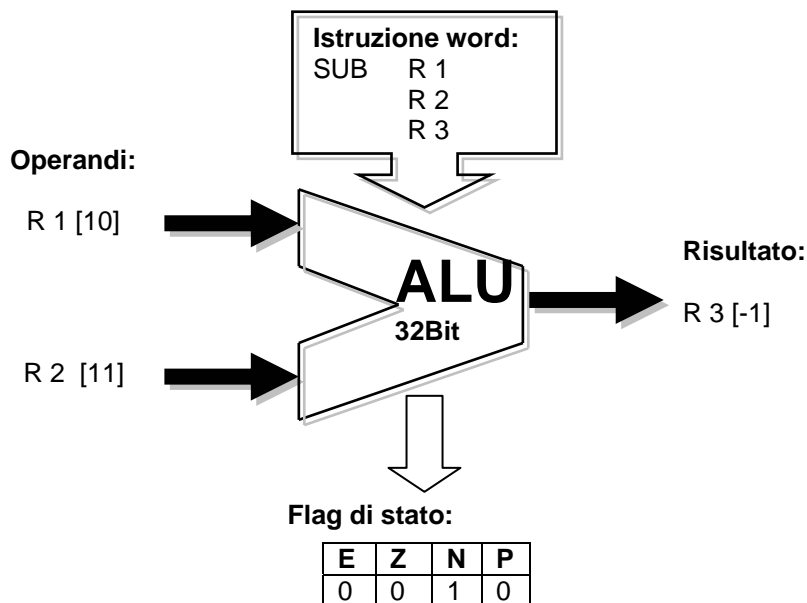
Radice quadrata per un numero negativo

Errore nell'assegnazione dell'interfaccia di comunicazione (istruzione SASI)

...

**Esempio:** Flag di stato dopo una sottrazione

Le flag di stato vengono impostate in base al risultato di una sottrazione ( $R 3 = R 1 - R 2$ ). Il valore dei registri vengono indicati tra parentesi quadre []. Se il risultato dell'operazione è negativo: viene impostata solo la flag N.

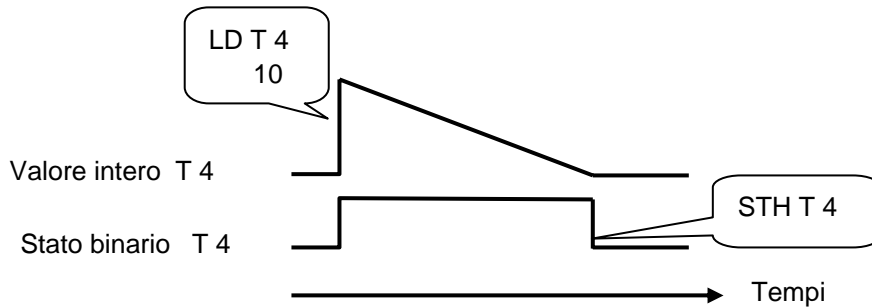


Se necessario, le flag di stato possono essere copiate nell'accumulatore per essere successivamente utilizzate con istruzioni binarie, istruzioni di salto programma oppure per la chiamata di PBs, FBs o SB:

ACC P	Copia lo stato della flag P nell'accumulatore
ACC N	Copia lo stato della flag N nell'accumulatore
ACC Z	Copia lo stato della flag Z nell'accumulatore
ACC E	Copia lo stato della flag E nell'accumulatore



### 7.5.5 Istruzioni di tipo Word per i temporizzatori



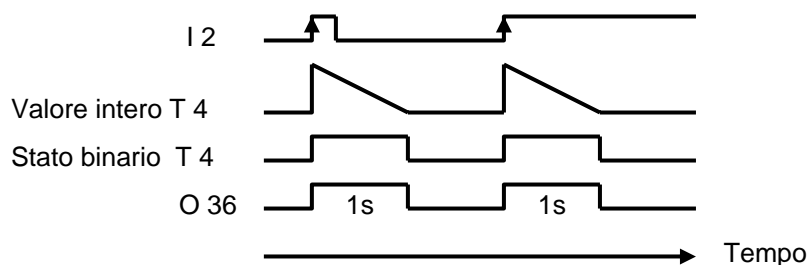
I temporizzatori contengono due valori: il valore intero del tempo di ritardo e lo stato binario del temporizzatore. Per implementare un tempo di ritardo, caricare il valore del tempo sotto forma di numero intero positivo che determinerà la lunghezza del tempo di ritardo in decimi di secondo<sup>1</sup>. Il controllore provvederà automaticamente a decrementare il valore del tempo fino a raggiungere il valore zero. Lo stato binario del temporizzatore, è alto fintanto che il valore del tempo si decrementa, e commuta a livello basso quando il temporizzatore raggiunge il livello zero.

<p><b>Caricamento di un tempo di ritardo</b></p> <pre>LD    T 4       10</pre> <p>Se lo stato dell'accumulatore è alto, il temporizzatore T4 verrà caricato con una costante di 10. Altrimenti il temporizzatore manterrà il valore attuale</p>	<p><b>Come leggere lo stato di un temporizzatore</b></p> <p>Usare una istruzione binaria:</p> <p>STH T 4 , ANH T 4, ORH T 4, ...</p>
---	--

Esempio:

Inviare un impulso di un secondo sull'uscita 36 ad ogni fronte di salita sull'ingresso 2.

Diagramma di stato:



Programma corrispondente:

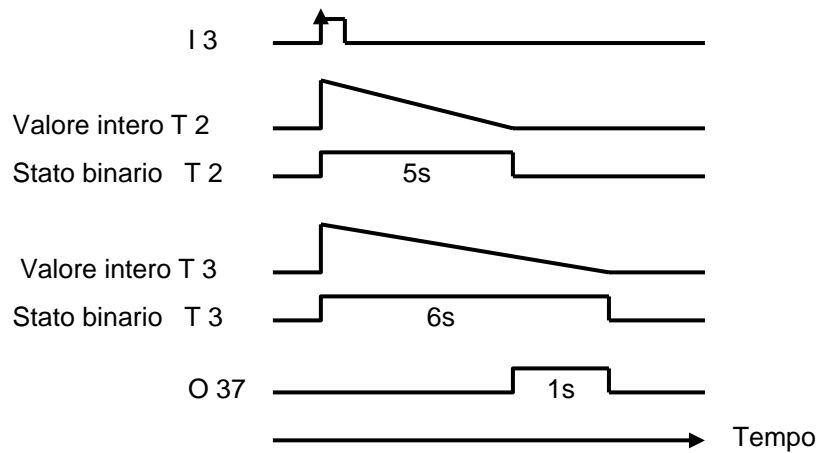
```
COB  0
      0
STH  I 2          ;Rilevamento del fronte di salita sull'ingresso 2 ...
DYN  F 2          ;...imposta lo stato dell'accumulatore a livello alto
LD   T 4          ;se accu è alto carica un ritardo equivalente a 10 unità di tempo
      10
STH  T 4          ;copia lo stato logico del ritardo di tempo sull'uscita 36
OUT  O 36
ECOB
```

<sup>1</sup> È possibile anche impostare una base tempi diversa da 1/10 di secondo (valore di default). Questo può essere effettuato dalla finestra *Software Settings (Impostazioni Software)*.

Esempio:

Inviare un impulso di un secondo sull'uscita 37 con un ritardo di 5 secondi ad ogni fronte di salita sull'ingresso 3

Diagramma di stato:



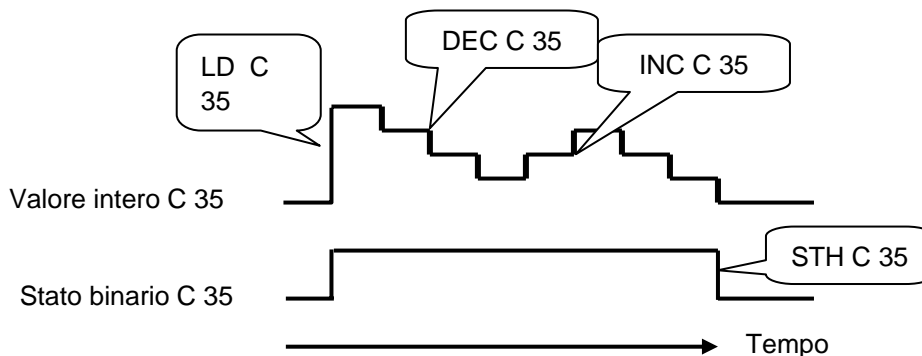
Programma corrispondente:

```

COB  0
      0
STH  I 3
DYN  F 3
LD   T 2
      50
LD   T 3
      60
STH  T 2
XOR  T 3
OUT  O 37
ECOB

```

### 7.5.6 Istruzioni per i contatori



Analogamente ai temporizzatori, anche i contatori hanno 2 valori: il valore intero del conteggio e lo stato binario del contatore.

Per implementare un conteggio, è sufficiente caricare un contatore con un valore intero positivo.

Contrariamente ai temporizzatori, i contatori possono solo essere incrementati o decrementati da istruzioni presenti nel programma utente. Lo stato binario di un contatore è alto quando il valore del conteggio è > di zero, e commuta a livello basso quando il contatore raggiunge il valore zero.

<p><b>Caricamento di un contatore</b>                  LD C 35                  10                  Se lo stato dell'accumulatore è alto, il contatore 35 verrà caricato con una costante di 10. Altrimenti il contatore manterrà il valore attuale.</p>	<p><b>Come leggere lo stato di un contatore</b>                   Usare una istruzione binaria:                   STH C 35, ANH C 35, ORH C 35, ...</p>
<p><b>Incremento di un contatore</b>                  INC C 35                  Se lo stato dell'accumulatore è alto, il contatore 35 verrà incrementato di una unità. Altrimenti il contatore manterrà il valore attuale.</p>	<p><b>Decremento di un contatore</b>                  DEC C 35                  Se lo stato dell'accumulatore è alto, il contatore 35 verrà decrementato di una unità. Altrimenti il contatore manterrà il valore attuale.</p>

**Esempio:** Conteggio degli impulsi provenienti da un ingresso con l'utilizzo di un contatore.

```

COB 0
      0
STH I 2 ; Copia lo stato dell'ingresso nell'accumulatore
DYN F 3 ; Forza l' accumulatore allo stato alto sul fronte positivo di I 2
INC C 35 ; Se lo stato dell'accumulatore è alto, incrementa il contatore
ECOB
    
```

Le istruzioni *STH* e *DYN* leggono le informazioni dall'ingresso 2 e posizionano l'accumulatore allo stato alto in caso di fronte positivo oppure allo stato basso in assenza di fronte. L'istruzione *INC* incrementerà l'accumulatore 35 in base allo stato dell'accumulatore.

### 7.5.7 Istruzioni dipendenti dall'accumulatore

Abbiamo visto che le istruzioni binarie fanno un largo uso dell'accumulatore, così come alcune istruzioni di tipo word.

Tuttavia, non tutte le istruzioni utilizzano l'accumulatore nello stesso modo. Esistono 7 istruzioni che lo utilizzano in modo speciale. Queste sono le cosiddette istruzioni dipendenti dell'accumulatore. Tali istruzioni vengono elaborate solo se l'accumulatore è stato precedentemente impostato a livello alto. Lo stato dell'accumulatore è quindi una condizione determinante.

Le 7 istruzioni dipendenti dall'accumulatore sono le seguenti:

SET	
RES	
COM	
LD	Solo per temporizzatori e contatori
LDL	Solo per temporizzatori e contatori
INC	Solo per temporizzatori e contatori
DEC	Solo per temporizzatori e contatori

Esempio:

Creare una base tempi che inverta lo stato di un'uscita una volta al secondo.

Questo esempio utilizza 3 istruzioni. La prima (*STL*) usa l'accumulatore per caricarvi lo stato invertito del temporizzatore. Le successive due istruzioni (*LD* e *COM*) dipendono dall'accumulatore. Esse caricheranno la base tempi e invertiranno lo stato dell'uscita, solo se l'accumulatore era stato precedentemente impostato a livello alto dall'istruzione *STL*.

```
COB  0
      0
STL  T 1 ; Se lo stato del temporizzatore è basso, lo stato dell'accumulatore sarà alto
LD   T 1 ; carica il tempo di ritardo con 10 unità di tempo
      10
COM  O 38 ; inverti lo stato dell'uscita
ECOB
```

### 7.5.8 Istruzioni di tipo word per operazioni aritmetiche su interi

Queste istruzioni sono utilizzate per il calcolo di equazioni aritmetiche usando registri e costanti in formato intero. Ogni istruzione aritmetica è costituita da diverse righe ed utilizza operandi come registri o costanti, il risultato verrà sempre caricato in un registro.

<b>Addizione</b>	<b>Sottrazione</b>	<b>Radice quadrata</b>
ADD R 0 R 1 R 3 ;R3=R0+R1	SUB R 0 K 18 R 3 ;R3=R0-18	SQR R 100 R 101
<b>Moltiplicazione</b>	<b>Divisione</b>	<b>Confronto</b>
MUL K 5 R 1 R 3 ;R3=5*R1	DIV R 0 R 1 R 3 ;R3=R0/R1 R 4 ;Resto	CMP R 0 R 1
<b>Incremento</b>	<b>Decremento</b>	<b>Inizializza registro</b>
INC R 0 ;R0= R0+1	INC R 0 ;R0= R0+1	LD R 0 K 19 ; R 0 = 19

#### Flag di stato

Tutte le istruzioni aritmetiche sopra elencate modificano le flag di stato in base al risultato dell'operazione (Positivo, Negativo, Zero, Errore), ad eccezione dell'istruzione usata per caricare un registro con una costante (LD).

#### Differenze tra registri e temporizzatori/contatori

Contrariamente per quanto avviene per i contatori, le istruzioni utilizzate per caricare una costante in un registro, incrementare o decrementare un registro non sono dipendenti dallo stato dell'accumulatore.

Il valore del registro che deve essere incrementato o decrementato può essere indifferentemente un numero intero positivo o negativo.

#### Esempio:

Confrontare il contenuto di due registri e commutare tre uscite in base alle seguenti condizioni:

Registri	O 32	O 33	O 34
R 0 > R 1	Alta	Bassa	Bassa
R 0 = R 1	Bassa	Alta	Bassa
R 0 < R 1	Bassa	Bassa	Alta

L'istruzione di confronto esegue una sottrazione R 0 – R 1 e imposta le flag di stato in base al risultato:

Registri	P	N	Z	E
R 0 > R 1	1	0	0	0
R 0 = R 1	1	0	1	0
R 0 < R 1	0	1	0	0

CMP R 0 ;Esegui la sottrazione R 0 – R 1, e modifica le flag di stato  
R 1 ; in base al risultato della sottrazione  
ACC P  
OUT O 32 ; R 0 > R 1

```

ACC  Z
OUT  0 33          ; R 0 = R 1
ACC  N
OUT  0 34          ; R 0 < R 1

```

### 7.5.9 Istruzioni di tipo word per operazioni aritmetiche in virgola mobile

Queste istruzioni sono utilizzate per il calcolo di equazioni aritmetiche usando registri e costanti in formato virgola mobile. Ogni istruzione aritmetica inizia con la lettera F per indicare che si tratta di una operazione in virgola mobile. Gli operandi di questa istruzione sono sempre dei registri, mai delle costanti. Nel caso in cui sia necessaria una costante, questa deve essere caricata in un registro e quindi può essere utilizzato tale registro nell'istruzione in virgola mobile.

<b>Addizione</b>	<b>Sottrazione</b>	<b>Radice quadrata</b>
FADD R 0 R 1 R 3 ;R3=R0+R1	FSUB R 0 R 1 R 3 ;R3=R0-R1	FSQR R 100 R 101 ;risultato
<b>Moltiplicazione</b>	<b>Divisione</b>	<b>Confronto</b>
FMUL R 0 R 1 R 3 ;R3=R0*R1	FDIV R 0 R 1 R 3 ;R3=R0/R1	FCMP R 0 R 1
<b>Seno</b>	<b>Coseno</b>	<b>Arco tangente</b>
FSIN R 10 R 11 ;risultato	FCOS R 10 R 11 ;risultato	FATAN R 10 R 11 ;risultato
<b>Esponente</b>	<b>Logaritmo Naturale</b>	<b>Valore assoluto</b>
FEXP R 20 R 21 ;risultato	FLN R 20 R 21 ;risultato	FABS R 30 R 31 ;risultato

Flag di stato

Tutte le istruzioni sopra elencate modificano le flag di stato, ad eccezione dell'istruzione *LD* per il caricamento di una costante in formato virgola mobile.

#### Inizializzazione di un registro

```

LD R 0
3.1415E0 ; R 0 = PI

```

### 7.5.10 Conversione di registri in formato intero e virgola mobile

Il PCD dispone di istruzioni separate per le operazioni aritmetiche su numeri interi e su numeri in virgola mobile. Se un programma applicativo deve sommare o moltiplicare due registri, uno contenente un numero intero e l'altro contenente un numero in virgola mobile, è necessario convertire entrambi i registri in formato intero o virgola mobile, prima di eseguire l'operazione aritmetica.

Conversione intero-virgola mobile	Conversioni virgola mobile-intero
IFP R 0 ; intero -> virgola mobile 0 ; esponente	FPI R 0 ;virgola mobile ->intero 0 ; esponente

### 7.5.11 Registro indice

Ogni COB dispone di un registro particolare: il registro indice. Il contenuto del registro indice può essere controllato con le seguenti istruzioni

<b>SEI K 10</b>	<b>SEt</b> Index register	Carica il registro indice con una costante di 10
<b>INI K 99</b>	<b>IN</b> crement Index register	Incrementa il registro indice e imposta accu allo stato alto fintanto che: Registro indice $\leq$ K 99
<b>DEI K 5</b>	<b>DE</b> crement Index register	Decrementa il registro indice e imposta accu allo stato alto fintanto che: Registro indice $\geq$ K 5
<b>STI R 0</b>	<b>ST</b> ore Index register	Copia il registro indice nel registro 0
<b>RSI R 0</b>	<b>Re</b> Store Index register	Copia il registro 0 nel registro indice

Molte istruzioni *PCD* supportano l'uso del registro indice. Questo registro permette l'indirizzamento indiretto di registri, flag, ingressi, uscite, temporizzatori ecc. utilizzati dalle istruzioni all'interno del programma. Tali istruzioni sono analoghe a quelle utilizzate normalmente, ma hanno una lettera *X* aggiuntiva.

Esempio:

I registri sono aree di memoria non volatili. Questo significa che mantengono la propria informazione anche dopo uno spegnimento o una ripartenza a freddo. Nel caso in cui desiderassimo rendere volatili un insieme di 100 registri, dovremmo inizializzare questi 100 registri con il valore 0 durante una ripartenza a freddo. Per inizializzare un registro a 0, possiamo utilizzare la seguente istruzione:

```
LD    R 10
      K 0
```

Nel caso in cui dovessimo inizializzare 100 registri (da R 10 a 109), dovremmo scrivere questa istruzione 100 volte, cambiando ogni volta l'indirizzo del registro. Questo risulterebbe alquanto fastidioso.

Un'altra soluzione potrebbe invece essere quella di inizializzare il registro indice con il valore zero ed implementare un loop di programma che provveda a caricare il primo registro con il valore zero, e incrementare il registro indice. In questo modo, ad ogni loop verrebbe caricato il valore zero in un registro diverso (R 10, R 11,.... R 109). Al 100° loop il registro indice raggiunge il valore massimo (K 99) e forza l'accumulatore allo stato basso. Questo permette di uscire dal loop ed elaborare il resto del programma.

```

      XOB          16          ;Blocco di ripartenza a freddo
      SEI          K 0          ;Indice = 0
LOOP: LDX         R 10          ;Carica indirizzo registro = 10 + indice
              0          ;con il valore zero
      INI          K 99         ;Incrementa indice e modifica lo stato
dell'accu
      JR           H LOOP      ;Se accu è alto, salta all'etichetta LOOP
      EXOB
      COB          0          ;Blocco ad organizzazione ciclica
              0
      ...
      ECOB
```

### 7.5.12 Salti di programma

Il repertorio di istruzioni IL dispone di tre istruzioni di salto programma. Queste permettono di elaborare una sequenza di istruzioni in base ad una condizione binaria, oppure implementare loop di programma per operazioni ripetitive (indicizzazione).

#### Istruzioni di salto (Jump)

<b>JR</b>	<b>Jump Relative</b>	Salta avanti o indietro di alcune righe a partire dalla riga contenente l'istruzione JR
<b>JPD</b>	<b>JumP Direct</b>	Salta ad un numero di riga specifico conteggiato a partire dall'inizio del blocco (COB,PB,...)
<b>JPI</b>	<b>JumP Indirect</b>	Come JPD, ma con il numero di riga contenuta in un registro

La destinazione del salto è generalmente indicata da una etichetta che definisce una riga del programma. Tuttavia, è possibile anche definire un salto relativo indicando il numero di righe da saltare in direzione avanti o indietro.

Salto utilizzando una etichetta di riga:      Salto utilizzando il numero di righe:

<pre>JR   L Next       INC R 10 Next: NOP</pre> 	<pre>JR   L +2       INC R 10       NOP</pre> 
---	---

Il salto deve sempre avvenire all'interno del blocco corrente (COB, PB,...), mai all'esterno.

Se necessario, il salto può essere implementato in modo incondizionato, oppure solo a fronte di una condizione binaria predefinita, quale lo stato dell'accumulatore o quello di una flag di stato.

#### Sintassi per una istruzione di salto incondizionato

Mnemonico	Etichetta	Descrizione
<b>JR</b>		Salto sempre eseguito alla riga corrispondente all'etichetta
<b>JPD</b>		
<b>JPI</b>		

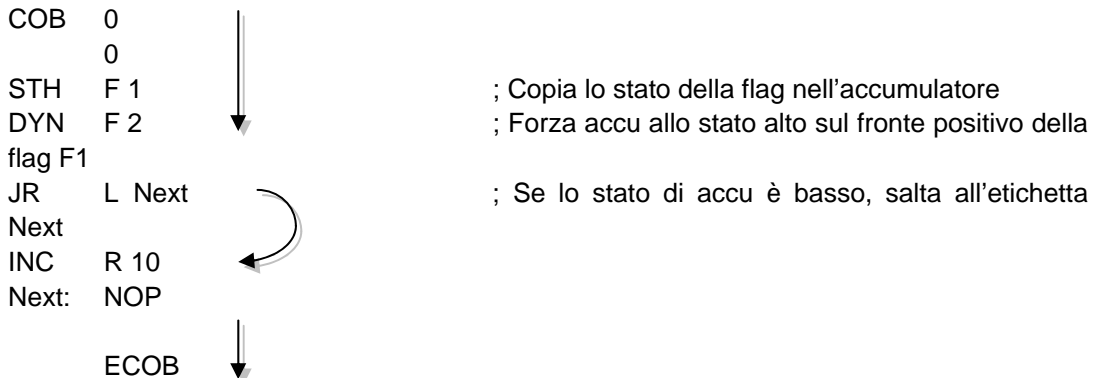
#### Sintassi per una istruzione di salto condizionato

Mnemonico	Condizione	Etichetta	Descrizione
<b>JR</b>	H		Solo se accu è alto
<b>JPD</b>	L		Solo se accu è basso
<b>JPI</b>	Z		Solo se la flag di stato Z è alta
	P		Solo se la flag di stato P è alta
	N		Solo se la flag di stato N è alta
	E		Solo se la flag di stato E è alta



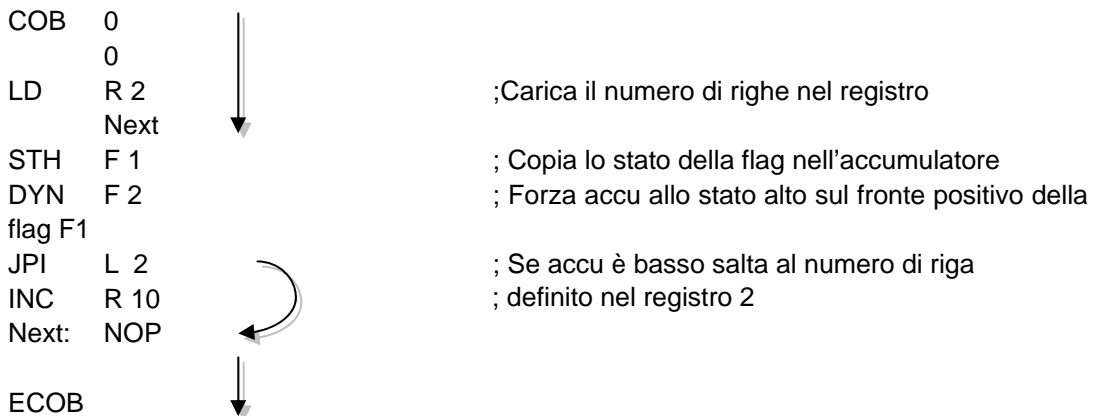
**Esempio:** Conteggiare gli impulsi provenienti da un ingresso binario utilizzando un registro (salto relativo)

Contrariamente a quanto avviene per i contatori, l'istruzione per incrementare un registro non dipende dallo stato dell'accumulatore. Può essere quindi conveniente utilizzare una istruzione di salto per incrementare un registro quando è necessario eseguire solo questa operazione.



Le istruzioni *STH* e *DYN* leggono l'informazione dalla flag F 1 e impostano l'accumulatore allo stato alto in caso di fronte positivo, oppure allo stato basso in assenza di un fronte. In base allo stato dell'accumulatore, l'istruzione *JR* provoca il salto alla riga corrispondente all'etichetta *Next*: oppure incrementa il registro con l'istruzione *INC*. La lettera *L* indica la condizione necessaria per eseguire il salto (in questo esempio, il salto verrà eseguito solo se l'accumulatore è allo stato basso).

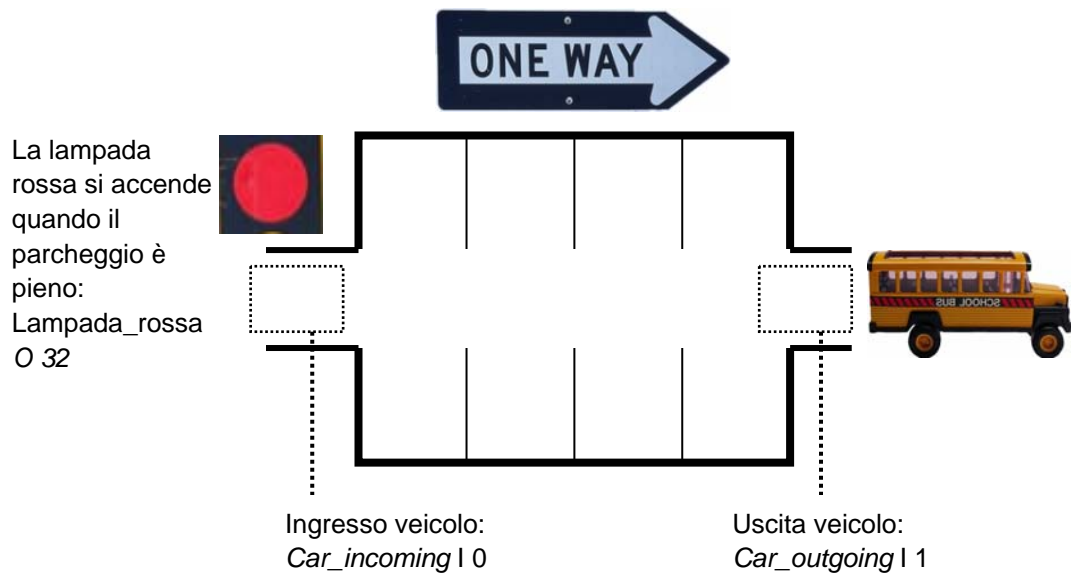
**Esempio:** Soluzione con salto indiretto



Il salto indiretto offre una notevole flessibilità. Il programma può infatti modificare autonomamente il numero di riga su cui eseguire il salto.

## 7.6 Scrittura del primo programma applicativo

Conteggiare il numero di posti liberi in un parcheggio ad 8 posti ed accendere una lampada rossa quando il parcheggio è pieno.

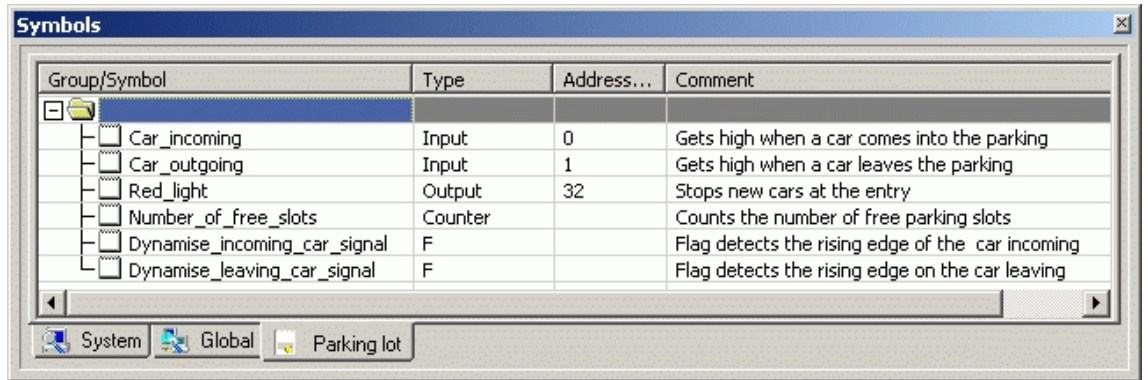


All'accensione del PCD, supponiamo che tutti i posti del parcheggio siano disponibili. Dovremo quindi iniziare il programma inizializzando il contatore dei posti liberi con il valore 8. Questa inizializzazione viene eseguita una sola volta, all'avvio del PCD. Provvederemo quindi a programmarla nel blocco di ripartenza a freddo: XOB 16. Le rimanenti funzioni del programma verranno eseguite da un blocco ad organizzazione ciclica (COB).

In entrata, il sensore *Car\_incoming* genera un impulso ogni qualvolta entra un nuovo veicolo. Il fronte di salita di questo segnale dovrà essere rilevato per decrementare il contatore dei posti liberi.

In uscita, un secondo sensore *Car\_outgoing* genera un impulso ogni qualvolta esce un veicolo. Il fronte di salita di questo segnale dovrà essere rilevato per incrementare il contatore dei posti liberi.

Quando il parcheggio è pieno, il valore intero del contatore indicherà zero posti disponibili. Lo stato logico del contatore indica questa situazione quando è basso. La lampada rossa posta all'ingresso del parcheggio dovrà quindi essere illuminata.



```

;-----
; Blocco organizzativo di ripartenza a freddo
;-----

```

```

XOB 16 ; Programma eseguito all'avvio
ACC H
LD Number_of_free_slots ; Inizializza il contatore dei posti liberi
8 ; con il valore 8 (in modo incondizionato)
EXOB ; Fine del programma iniziale

```

```

;-----
; Blocco ad organizzazione ciclica
;-----

```

```

COB 0 ; Programma ciclico
0 ; Nessun tempo di supervisione

STH Car_incoming ; Ingresso di un veicolo nel parcheggio:
DYN Dynamise_incoming_car_signal ; Sul fronte positivo del segnale di ingresso veicolo
DEC Number_of_free_slots ; Decrementa il numero dei posti liberi

```

```

;-----

```

```

STH Car_outgoing ; Uscita di un veicolo nel parcheggio:
DYN Dynamise_leaving_car_signal ; Sul fronte positivo del segnale di uscita veicolo
INC Number_of_free_slots ; Incrementa il numero dei posti liberi

```

```

;-----

```

```

STL Number_of_free_slots ; Se non vi sono posti liberi (stato cont. = basso)
OUT Red_light ; Accendere la lampada rossa

```

```

ECOB ; Fine del programma ciclico

```

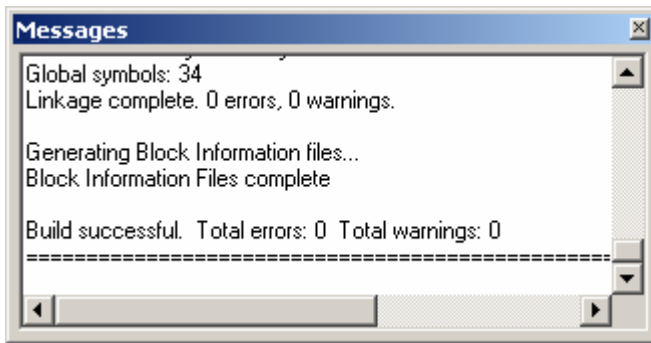
## 7.7 Costruzione (Build) del programma



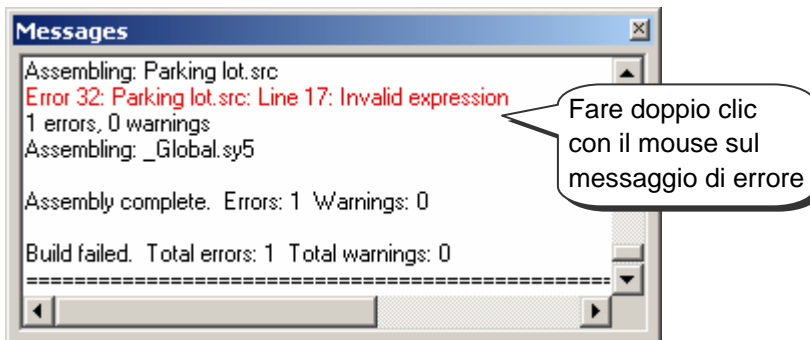
Costruisci  
Tutto

Il programma utente è stato completamente scritto, ma non è ancora eseguibile dal PCD. Deve infatti essere prima tradotto in un file binario. Questo verrà eseguito dal pacchetto di programmazione quando l'utente attiva il menu *CPU Build (CPU Costruisci)*, o preme il pulsante *Build (Costruisci)* nel Project Manager o nell'editore IL.

La finestra Messages (Messaggi) indica lo stato di avanzamento del processo di costruzione. Si potrà notare che l'operazione di costruzione prevede due stadi: Assemblaggio e Link. Se il programma è stato scritto correttamente, l'operazione di costruzione terminerà con il messaggio *Build successful. Total errors 0 Total warnings: 0 (Costruzione terminata con successo. Totale errori 0, Totale avvertimenti: 0)*.



Gli eventuali errori verranno indicati con un messaggio scritto in rosso. Facendo doppio clic con il mouse su tale messaggio si abilita la localizzazione dell'errore nel programma applicativo.



L'errore è indicato in rosso oppure con una freccia

<b>STH</b>	<b>Car_incoming2</b>
<b>DYN</b>	<b>Dynamise_incoming_car_signal</b>
<b>DEC</b>	<b>Number_of_free_slots</b>

Correzione dell'errore

<b>STH</b>	<b>Car_incoming</b>
<b>DYN</b>	<b>Dynamise_incoming_car_signal</b>
<b>DEC</b>	<b>Number_of_free_slots</b>

## 7.8 Trasferimento (Download) del programma nel PCD



Trasferimento  
Programma

Il programma applicativo a questo punto è pronto. Ora dovrà essere trasferito dal computer al PCD utilizzando il comando di menu *Online, Download Program (Online, Trasferimento Programma)*, oppure il pulsante *Trasferimento Programma* nella barra degli strumenti della finestra SAIA Project Manager.

Nel caso in cui si verifichi un qualsiasi problema di comunicazione, controllare nuovamente le configurazioni in *Settings Online (Impostazioni Online)* e *Settings Hardware (Impostazioni Hardware)* e inoltre controllare il cavo di comunicazione tra computer e PCD. (PCD8.K111, USB)

## 7.9 Debug di un programma

I programmi non sono sempre perfetti nella loro prima versione. Può quindi essere utile testarli attentamente. Il test di un programma è supportato dallo stesso editore utilizzato per scriverlo.

### 7.9.1 Visualizzazione del codice compilato



Visualizza/  
Nascondi  
Codice

Il menu *View Code (Visualizza Codice)*, oppure il pulsante *Visualizza/Nascondi Codice*, permettono di visualizzare su una singola pagina IL il codice sorgente e il codice ottenuto dopo il processo di costruzione.

Le righe bianche rappresentano il codice sorgente originale, con i relativi simboli e commenti.

Le righe grigie rappresentano invece il codice prodotto dal processo di costruzione (build), con gli indirizzi degli operandi e i numeri delle righe di programma.

```

Parking lot.src
; Cyclical Organisation Block
;-----
          COB      0          ; Cyclical program
          COB      0          ; No supervision time
000007 COB      0
000008 COB      0
000010 NOP

          STH      Car_incoming |          ; A car comes into the parking:
000011 STH      I|0 0
          DYN      Dynamise_incoming_car_signal ; On the positiv flank of incoming si
000012 DYN      F 7502
          DEC      Number_of_free_slots      ; Decrement the number of free park:
000013 DEC      C 1400

;-----

          STH      Car_outgoing          ; A car leaves into the parking:
000014 STH      I|0 1
          DYN      Dynamise_leaving_car_signal ; On the positiv flank of outgoing si
000015 DYN      F 7503

          INC      Number_of_free_slots      ; Increment the number of free park:
000016 INC      C 1400
    
```

## 7.9.2 Pulsanti On/Offline - Run – Stop – Step by step

La modalità In Linea permette di comunicare con il PCD per controllare la modalità di funzionamento (Run, Stop, Step-by-step) (Esecuzione, Stop, Passo-passo). Può anche essere visualizzata qualsiasi informazione necessaria per il test del programma.

Premere il pulsante *In linea/Fuori linea*

Porre il controllore in modalità run premendo il pulsante *Run (Esecuzione)*



Contemporaneamente, osservare il led RUN sul coperchio del PCD. Quando si preme il pulsante *Run* si deve accendere. Questo significa che il PCD sta eseguendo il programma utente.

Quando si preme il pulsante *Stop*, il led *RUN* si spegne. Questo significa che il PCD ha interrotto l'esecuzione del programma utente.



Dopo aver premuto il pulsante *Stop*, osservare la riga visualizzata in rosso. Questa indica l'istruzione in cui il programma si è arrestato. Il numero tra parentesi quadra indica il valore intero del contatore 1400. Quindi, a destra di tale numero vengono visualizzati gli stati dell'accumulatore, delle flag di stato e del registro indice.

```

DYN Dynamise_leaving_car_signal ; On the positiv flank of outgoing signal
000015 DYN F 7503
INC Number_of_free_slots ; Increment the number of free parking slots
000016 INC C 1400 [8] A0 20 N0 P1 E0 IX0000

```

### 7.9.3 Modalità passo-passo



Esegui fino al cursore

Se il PCD è in modalità "Run", contrassegnare la prima riga che si vuol osservare in modalità passo-passo e premere il pulsante *Esegui fino al cursore*.

Il PCD si arresta quando raggiunge la riga indicata dal cursore. Per iniziare l'esecuzione del programma in modalità passo-passo, premere il tasto F11, oppure uno dei pulsanti qui di seguito indicati.

Se il programma chiama un qualsiasi blocco PB, FB o SB, non è sempre necessario eseguire questi ultimi in modalità passo-passo. Sono disponibili le seguenti tre opzioni:



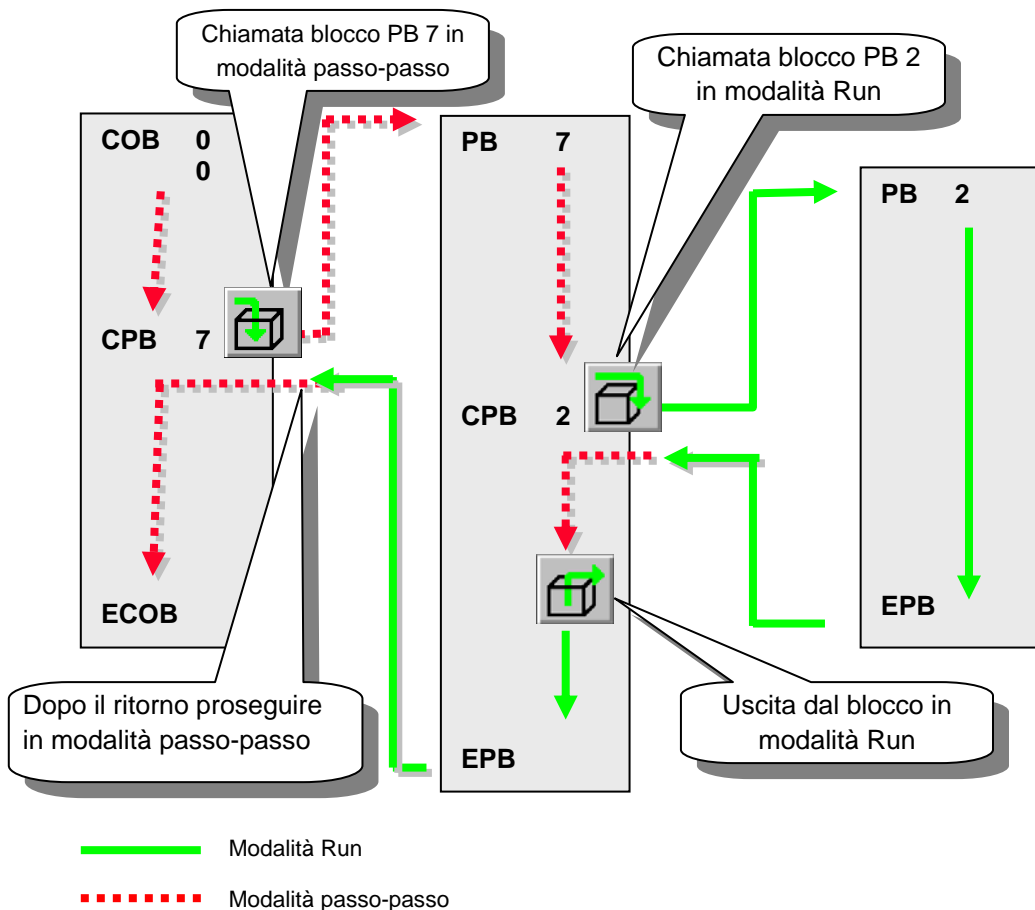
Entrare nel blocco ed eseguirlo in modalità passo-passo.



Elaborare il blocco chiamato in modalità Run, quindi proseguire in modalità passo-passo dopo il ritorno al blocco che aveva effettuato la chiamata.



Se il programma è entrato in un blocco il cui contenuto è irrilevante ai fini del test, è possibile uscire rapidamente da tale blocco in modalità Run e quindi proseguire in modalità passo-passo dopo il ritorno al blocco che aveva effettuato la chiamata.



```

STH Car_outgoing ; A car leaves into the parking:
000014 STH I|O 1 [0] A0 20 N0 P1 E0 IX0000
    
```

Per ogni passo di programma, notare la riga visualizzata in rosso. Questa si sposta progressivamente sulla riga di istruzione successiva. Il numero visualizzato tra parentesi quadre rappresenta lo stato logico dell'ingresso I 1. A destra di tale numero vengono visualizzati gli stati dell'accumulatore, delle flag di stato e del registro indice.

## 7.9.4 Breakpoint (Interruzioni)

I Breakpoint (Interruzioni) permettono di arrestare l'esecuzione di un programma al verificarsi di un evento collegato ad una riga di programma a un simbolo:

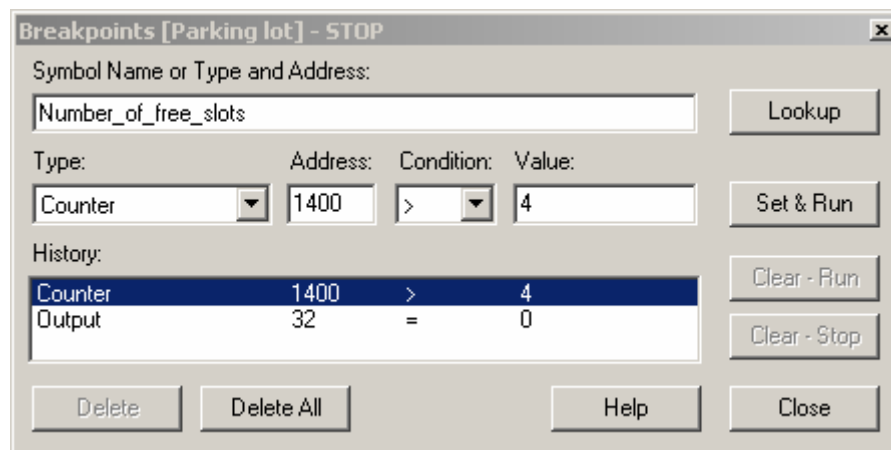
- Stato alto o basso di un ingresso, uscita, flag o flag di stato
- Valore presente in registri o contatori

“Breakpoint” su simbolo



Imposta/Annulla  
Breakpoint

La condizione di breakpoint può essere definita con l'aiuto del menu *Online Breakpoints*, oppure utilizzando il pulsante *Imposta/Annulla Breakpoint*.



Servendosi della finestra di dialogo sopra raffigurata, definire il tipo di simbolo (Type) ed il relativo indirizzo/numero (Address) oppure trascinare semplicemente un simbolo dall'editore di simboli al campo *Symbol Name (Nome Simbolo)*, quindi definire la condizione di interruzione (Condition) ed il Valore/Stato desiderato (Value).

Il PCD può essere forzato in modalità “*Run condizionato*” utilizzando il pulsante *Set&Run (Imposta e Passa in Run)*. Il LED *Run* del PCD inizia a lampeggiare e il pulsante *Run* si accende alternativamente con colore verde e rosso.

Quando viene raggiunta la condizione di breakpoint il PCD va automaticamente in modalità stop. Nell'esempio sopra illustrato, questo si verifica quando una istruzione modifica il valore del contatore 1400, con un valore maggiore di 4. La riga successiva all'ultima istruzione elaborata dal PCD verrà visualizzata in rosso. A questo punto è possibile proseguire l'elaborazione del programma in modalità passo-passo, oppure con un'altra condizione di breakpoint.

- Il pulsante *Clear-Run (Annulla-Run)* forza il PCD in modalità RUN. Il LED *Run* si illuminerà con luce fissa ed il pulsante di *Run* verrà visualizzato in colore verde
- Il pulsante *Clear-Stop (Annulla-Stop)* forza il PCD in modalità STOP. Il LED *Run* si spegnerà ed il pulsante di *Run* verrà visualizzato in colore rosso.

Se si sono definite più interruzioni (breakpoint) condizionali, esse verranno inserite nel campo *History (Storico)*. In questo caso, è possibile selezionare con il mouse una qualsiasi interruzione predefinita ed attivarla con il pulsante *Set & Run (Imposta e Passa in Run)*.

“Breakpoint” su riga di programma

Selezionare la riga di programma desiderata quindi selezionare la voce di menu *Online, Run to Cursor (In Linea, Esegui Fino a Corsore)* per impostare l'interruzione del programma al raggiungimento dalla riga di programma interessata. A questo punto, procedere con l'esecuzione del programma in modalità passo-passo.

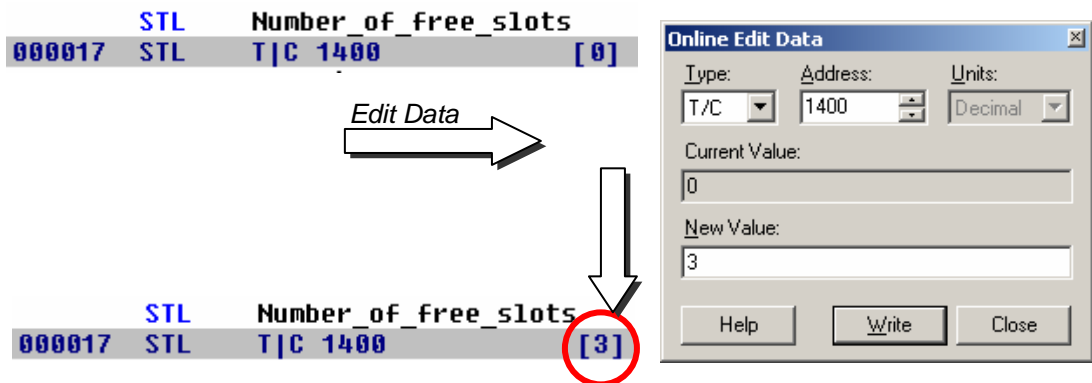


### 7.9.5 Modifica Online del programma

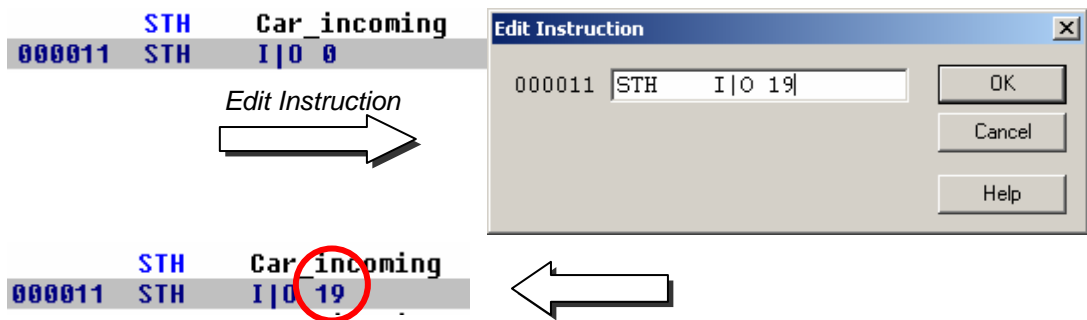
Quando si esegue il test di un programma in modalità passo-passo, può essere utile modificare gli stati/valori di determinati operandi/simboli e controllare il comportamento del programma al verificarsi di certe condizioni.

Selezionare una delle righe attive (in grigio) utilizzando il mouse e fare clic con il tasto destro per visualizzare il menu contestuale.

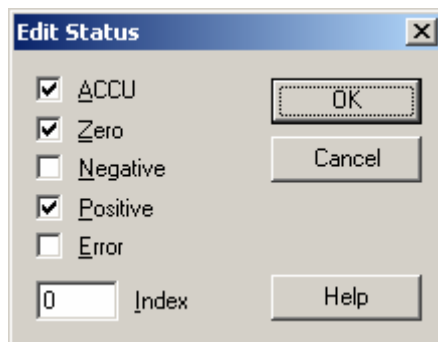
La voce del menu contestuale *Edit Data (Modifica Dati)* permette di modificare lo stato/valore di un operando dell'istruzione selezionata.



La voce del menu contestuale *Edit Instruction (Modifica Istruzione)* permette di modificare il codice mnemonico e l'indirizzo dell'operando della riga dell'istruzione selezionata.



Anche le flag di stato possono essere modificate con l'aiuto della voce del menu contestuale *Edit Status (Modifica Stato)*.



## 7.9.6 Visualizzazione e modifica degli stati dei simboli con la “Finestra di Monitoraggio” (*Watch Window*)



Finestra di Monitoraggio

Un altro utile modo per testare e visualizzare lo stato dei simboli nel nostro esempio di parcheggio è quello fornito dalla *Watch Window* (*Finestra di Monitoraggio*). Premere il pulsante di attivazione dedicato del SAIA Project Manager quindi trascinare i simboli dall'editore dei simboli alla Finestra di Monitoraggio.

1. Posizionare il cursore del mouse al centro dell'icona del simbolo e premere il tasto sinistro

2. Tenere premuto il tasto del mouse e trascinare il simbolo nella “*Watch Window*” (*Finestra di Monitoraggio*)

3. Simboli con i relativi commenti e stati/valori

Symbol	Address	Value	Modify Value	Symbol Comment
Car_incoming	I 0	0		Gets high when a car comes i...
Car_outgoing	I 1	0		Gets high when a car leaves t...
Red_light	O 32	0		Stops new cars at the entry
Number_of_free_slots	C 1400	8		Counts the number of free p...

Per modificare lo stato/valore di uno dei simboli nella finestra, operare come segue:

-  1. Avvio/Arresto Monitoraggio

Symbol	Address	Value	Modify Value	Symbol Comment
Car_incoming	I 0	0		Gets high when a car comes i...
Car_outgoing	I 1	0		Gets high when a car leaves t...
Red_light	O 32	0		Stops new cars at the entry
Number_of_free_slots	C 1400	8	5	Counts the number of free p...

2. Posizionare il cursore del mouse sul valore da modificare.  
Fare doppio clic con il tasto sinistro del mouse ed inserire il nuovo valore.



3. Trasferire i valori

## 7.10 Messa in servizio di un modulo analogico

Tutte le istruzioni di programma presentate fino ad ora hanno utilizzato ingressi o uscite digitali, con l'inserimento degli indirizzi o dei simboli come operandi.

Esempio: ANH I 45

Con i moduli di ingressi o di uscite analogiche, deve essere utilizzata, comunque, una routine di acquisizione dei valori per i moduli analogici. Ci sono routine diverse per i diversi tipi di moduli analogici. Le descrizioni si possono trovare nel manuale hardware del vostro PCD.

### 7.10.1 Esempio per i moduli di ingressi analogici PCD2.W340

Se il PCD è equipaggiato con un modulo PCD2.W340, che è fornito di 8 canali di ingresso universali, deve essere utilizzata la seguente routine:

BA	EQU	O 96	; Indirizzo di base del modulo nel PCD
	ACC	H	; ACCU deve essere alto
	LD	R 100	; Definizione del canale di misura ( 0...7)
		2	
	MUL	R 100	
		K 32	; Calcolo
		R 100	; del byte di controllo
	ADD	R 100	; richiesti per
		K 264	; la selezione del canale.
		R 100	
	SET	BA+15	; Avvio della conversione A/D
	BITO	9	; Invio del
		R 100	; byte di controllo
		BA+0	; a W3xx
	BITIR	12	; Lettura dei 12 bit della misura (0...4095) nel R
77		BA+0	
		R 77	
	RES	BA+15	; Fine della conversione A/D

Il PCD2.W340 è un modulo universale. Supporta misure nei campi 0..10V, 0..2.5V, 0..20 mA e sensori di temperatura Pt/Ni 1000. Un ponticello sul modulo permette di selezionare il campo di misura. La risoluzione è di 12 bit, equivalenti a 4095 distinti stati di misura.

La routine precedente seleziona il canale definito nel registro 100 e fornisce una misura non ingegnerizzata nel registro 77. Questo modulo con risoluzione di 12 bit, fornisce un valore di misura tra 0 e 4095.

L'utente ha il compito di convertire la misura in una unità fisica standard.

### 7.10.2 Esempio per i moduli di uscite analogiche PCD2.W610

Le uscite analogiche funzionano in modo simile agli ingressi.

Se il PCD è equipaggiato con un modulo PCD2.W610, che è fornito di 4 canali di uscita universali, deve essere usata la seguente routine:

```

BA EQU O 96 ; Indirizzo di base del modulo inserito nel PCD
ACC H ; ACCU deve essere alto
LD R 100 ; Definizione del canale di uscita ( 0...6)
2
BITOR 2 ; Trasferimento del canale a W6x0
R 100
BA+0
BITOR 2 ; Scrive 2 bit
R 100
BA+0
LD R 277 ; Definizione del valore digitale dell'uscita (
0...4095)
3879
BITO R 12 ; Trasferimento dei 12 bit del valore di uscita nel
W6x0
R 277
BA+0
SET BA+12 ; Avvio della conversione D/A

```

Un ponticello sul modulo permette di selezionare il campo delle uscite: 0...20 mA o 0...10 V. La risoluzione è di 12 bit, che corrispondono a 4095 distinti valori di stato.

Il valore intero nel registro 12 determina la tensione o la corrente di uscita sul canale definito nel registro 100:

Valore nel registro 12	Tensione d'uscita [V]	Corrente d'uscita [mA]
0	0	0
2047	5	10
4095	10	20



Per informazioni dettagliate ed esempi di programmazione per moduli analogici in Lista Istruzioni (IL), fare riferimento al manuale hardware o all'indirizzo internet:

<http://www.sbc-support.ch>

## Indice

---

8	Strumenti Aggiuntivi	8-3
8.1	Introduzione	8-3
8.2	Utility di Trasferimento Dati	8-4
8.2.1	Utilizzo del programma di trasferimento dati (Data Transfer)	8-4
8.2.2	Avvio del Trasferimento Dati	8-4
8.2.3	Salvataggio dei dati con la finestra Trasferimento Dati Veloce (Quick Data Upload)	8-4
8.2.4	Ripristino dei dati	8-5
8.2.5	Salvataggio dei dati con l'aiuto del file "Script"	8-5
8.2.6	Ripristino dei dati con l'aiuto del file "Script"	8-6
8.2.7	Opzioni di trasferimento dal PCD	8-6
8.2.8	Salvataggio dei dati con la modalità riga di comando	8-7
8.3	"Finestra di Monitoraggio" (Watch Window)	8-8
8.3.1	Apertura della "Finestra di Monitoraggio" (Watch Window)	8-8
8.3.2	Aggiunta di dati ad una "Finestra di Monitoraggio" (Watch Window)	8-9
8.3.3	Visualizzazione in linea dei dati	8-10
8.3.4	Modifica in linea dei dati	8-10
8.3.5	Formato di visualizzazione	8-10
8.3.6	"Finestra di Monitoraggio" (Watch Window) ed applicazioni multi-CPU	8-11
8.4	Configurazione	8-12
8.4.1	Configuratore Offline (Fuori Linea)	8-12
8.4.2	Configuratore Online (In Linea)	8-12
8.4.3	Finestra del Configuratore Online	8-12
8.4.4	Regolare l'orologio del PCD	8-13
8.4.5	Tabella Storica (History)	8-13
8.5	Programmazione di memorie EPROM	8-14
8.6	Aggiornamento del firmware (Firmware Downloader)	8-15
8.7	Menu utente	8-16



## **8 Strumenti Aggiuntivi**

---

### **8.1 Introduzione**

PG5 prevede una serie di programmi di utilità aggiuntivi per una varietà di servizi.

## 8.2 Utility di Trasferimento Dati

### 8.2.1 Utilizzo del programma di trasferimento dati (Data Transfer)

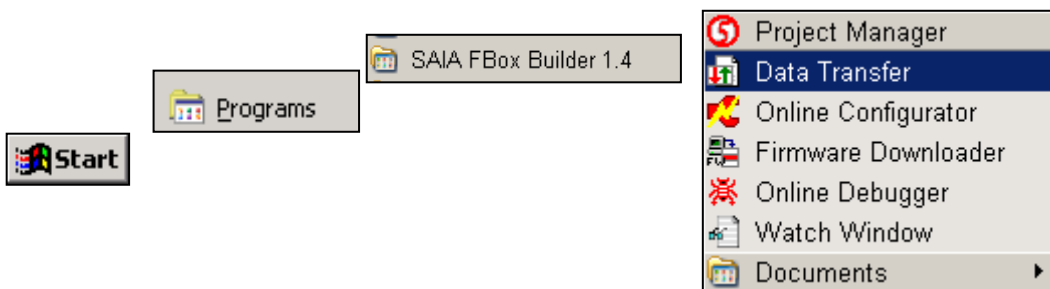
Questo programma può essere utilizzato per salvare gli stati/valori del PCD in un file ASCII (\*.dt5) oppure per ripristinare tali dati dal file alla memoria del PCD. Con questo programma si possono trasferire i seguenti dati: ingressi, uscite, flag, temporizzatori, contatori, registri, blocchi di dati e di testo.

Attenzione! Con l'utility *Data Transfer* non vengono salvati il programma del PCD e le configurazioni hardware. Per salvare il programma, le configurazioni hardware e i dati, è consigliabile effettuare il back up del programma. Vedere la descrizione del Project Manager.

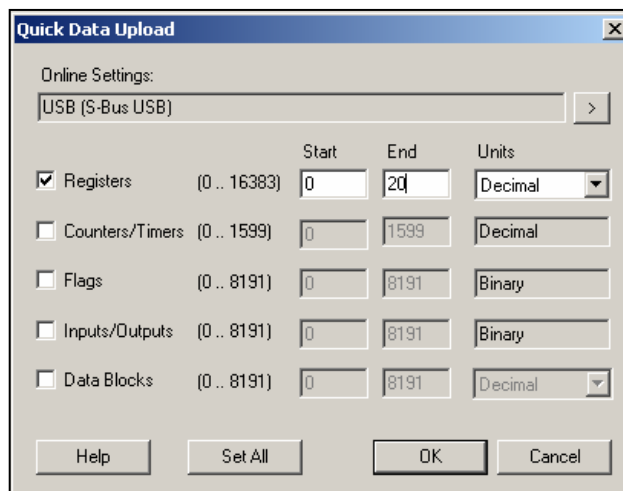
### 8.2.2 Avvio del Trasferimento Dati

Avviare il programma utilizzando il menu:

Start --> Programs --> SAIA PG5 V 1.4 --> Data Transfer



### 8.2.3 Salvataggio dei dati con la finestra Trasferimento Dati Veloce (Quick Data Upload)



Trasferimento  
Dati Veloce

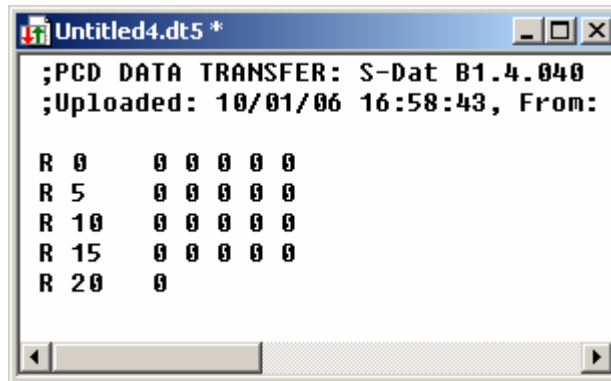
Selezionare la voce di menu *Online, Quick Data Upload ... (Online, Trasferimento Dati Veloce)* o premere il pulsante *Trasferimento Dati Veloce* per visualizzare la finestra sopra raffigurata. Selezionare i tipi di dati da salvare, il campo di indirizzi, e possibilmente anche il formato di visualizzazione dei registri. Premere il pulsante OK per iniziare il trasferimento dati.

Se viene visualizzato un messaggio simile a quello raffigurato a destra, controllare i parametri della comunicazione utilizzando il comando di menu *Online, Settings Online (Online, Impostazioni Online)* e verificare che PC e PCD siano correttamente collegati con il cavo PCD8.K111.





Il trasferimento dei dati richiede alcuni istanti, dopodiché vengono visualizzate le seguenti informazioni:



```

;PCD DATA TRANSFER: S-Dat B1.4.040
;Uploaded: 10/01/06 16:58:43, From:

R 0      0 0 0 0 0
R 5      0 0 0 0 0
R 10     0 0 0 0 0
R 15     0 0 0 0 0
R 20     0

```

Il file di dati può essere modificato inserendo nuovi valori, quindi può essere salvato utilizzando il comando di menu *File, Save (File, Salva)*, oppure il pulsante *Salva* nella barra degli strumenti.

### 8.2.4 Ripristino dei dati



*Apri*

I file precedentemente salvati possono essere nuovamente visualizzati utilizzando il comando di menu *File, Open (File, Apri)* oppure il pulsante *Apri* nella barra degli strumenti.

Se necessario, l'utente può modificare i valori contenuti nel file.



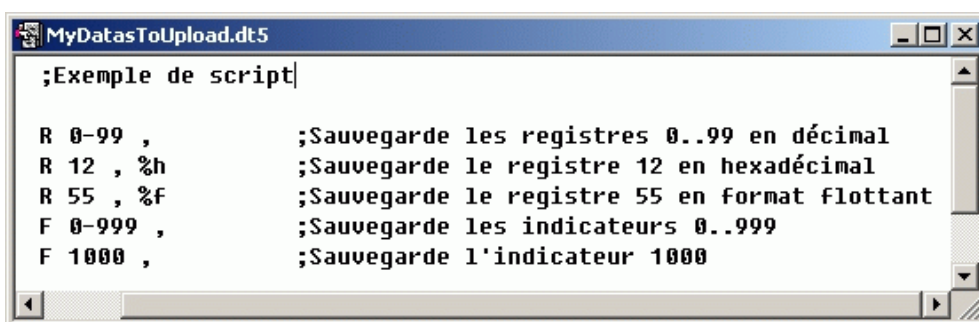
*Trasferimento  
a PCD*

Il ripristino dei dati nella memoria del PCD viene effettuato con il comando di menu *Online, Download Data to the PCD (Online, Trasferimento Dati al PCD)*, oppure con il pulsante *Trasferimento a PCD*.

### 8.2.5 Salvataggio dei dati con l'aiuto del file "Script"

Se necessario, è possibile creare l'elenco dei dati da salvare in un file script.

Esempio:



```

;Exemple de script

R 0-99 ,           ;Sauvegarde les registres 0..99 en décimal
R 12 , %h         ;Sauvegarde le registre 12 en hexadécimal
R 55 , %f         ;Sauvegarde le registre 55 en format flottant
F 0-999 ,         ;Sauvegarde les indicateurs 0..999
F 1000 ,         ;Sauvegarde l'indicateur 1000

```



*Trasferimento  
da PCD*

Selezionare il comando di menu *Online, Upload Data from PCD ... (Online, Trasferimento Dati da PCD ...)* oppure il pulsante *Trasferimento da PCD*, per trasferire i dati in una seconda finestra distinta dalla finestra di controllo.

Per maggiori informazioni sui comandi script disponibili si prega di fare riferimento all'help del programma. Vedere il comando del menu *Help, Help Topics F1, General (Aiuto, Argomenti di Aiuto F1, Generalità)*.

### 8.2.6 Ripristino dei dati con l'aiuto del file "Script"

Un file "script" permette anche di modificare i dati che si desidera ripristinare. Esempio:



Trasferimento a PCD

```

;Exemple de script
R 0-99, 0 ;Charge les registre 0 à 99 avec zéro
R 12, 32h ;Charge le registre 12 avec 32 hexadecimal
R 55, 64.3 ;Charge le registre 55 avec 64.3 flottant
F 0-999, 0 ;Charge les indicateurs 0 à l'état bas
F 1000, 1 ;Charge l'indicateur 1000 à l'état haut

```

Selezionare il comando di menu *Online, Download Data to PCD ... (Online, Trasferimento Dati al PCD ...)* oppure il pulsante *Trasferimento a PCD* per trasferire i dati del file script al PCD.

### 8.2.7 Opzioni di trasferimento dal PCD

La finestra visualizzata con il comando di menu *Edit, Options (Modifica, Opzioni)* permette di adattare il formato dei dati da salvare nel file '\*.dt5'.

Utilizzando le seguenti opzioni, un file di dati può essere facilmente importato nell'ambiente *Microsoft Excel*.

The image shows two dialog boxes and a sample data file. The **Quick Data Upload** dialog has the following settings:

	Start	End	Units
<input checked="" type="checkbox"/> Registers (0 .. 4095)	0	2	Decimal
<input type="checkbox"/> Counters/Timers (0 .. 1599)	0	1599	
<input checked="" type="checkbox"/> Flags (0 .. 8191)	0	6	
<input type="checkbox"/> Outputs/Inputs (0 .. 8191)	0	8191	
<input type="checkbox"/> Data Blocks (0 .. 7999)	0	7999	

The **Options** dialog has the following settings:

- Address separator: TAB
- Data separator: TAB
- For R / T / C / DB: 2
- For I / O / F: 4
- Formatting:
  - Empty line between different data types
  - Type and address at beginning of line
  - Data separator at end of line

The sample data file **Untitled1.dt5** contains the following content:

```

;PCD DATA TRANS
;Uploaded: 11/01/06 08:51:22, From: PCD3.M33x 018, :
R 0 0 0
R 2 0
F 0 0 0 0 0
F 4 0 0 0

```

Annotations in the image explain the format:

- Tipo e indirizzo all'inizio della riga**: Points to the first column of each line (R, F).
- Riga vuota tra tipi di dati diversi**: Points to the empty line between the register and flag sections.
- Separatore indirizzi**: Points to the space between the type and the address.
- Separatore dati**: Points to the space between the data values.
- Valori per riga**: Points to the data values themselves.

## 8.2.8 Salvataggio dei dati con la modalità riga di comando

Il programma *Data Transfer (Trasferimento Dati)* può essere gestito anche con l'ausilio delle righe di comando DOS.

Questo permette di creare dei file batch per il salvataggio automatico dei dati PCD, ad intervalli regolari. I dati potranno quindi essere utilizzati da un programma Microsoft Excel o database, ...

### Sintassi della riga di comando:

*SDAT [Nome\_del\_file[.dt5][dati...]][/R=nnn]/I0nnn[/A=nnn]/D=nnn]*

Nome_del_file	Nome del file da salvare/ripristinare
Dati...	Definizione dei dati da salvare. Se non viene definito alcun dato, verrà ripristinato nel PCD l'intero file. <i>Formato: &lt;tipo&gt;&lt;inizio&gt;[-&lt;fine&gt;][unità]</i>
tipo	R,C,O,F,DB (C= contatori/temporizzatori, O = ingressi/uscite)
inizio	Primo indirizzo
fine	Ultimo indirizzo
unità	D,H,F (Decimale, esadecimale, virgola mobile) per R,C,DB
/R=nnn	nnn = valore linea per R,T,C,DB ( 1..256, default = 5)
/I =nnn	nnn = valore linea per I,O,F ( 1..256, default = 10)
/A=nnn	nnn = separatore indirizzi (TABULAZIONE, SPAZIO, VIRGOLA, DUE PUNTI, default= TABULAZIONE)
	nnn = separatore indirizzi (TABULAZIONE, SPAZIO, VIRGOLA, DUE PUNTI, default= TABULAZIONE)
/D=nnn	PUNTI, default= TABULAZIONE)

### Esempio:

sdat5 Mieidati.dt5 R0-99 R12H R55F F0-999 F1000 /R005 /I010

### 8.3 “Finestra di Monitoraggio” (*Watch Window*)

La *Finestra di Monitoraggio (Watch Window)* è uno strumento prezioso per il controllo dei programmi e delle installazioni. Essa permette infatti di visualizzare e modificare “online” tutti i dati di una applicazione.

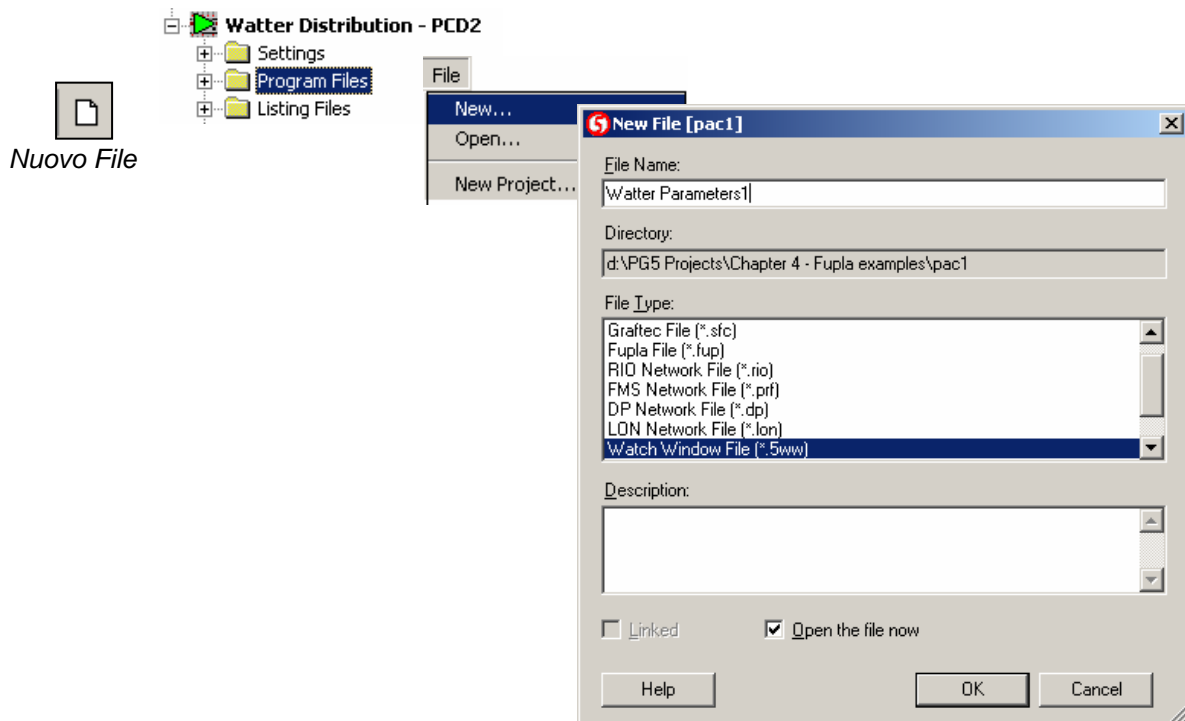
#### 8.3.1 Apertura della “Finestra di Monitoraggio” (*Watch Window*)



*Finestra di Monitoraggio*

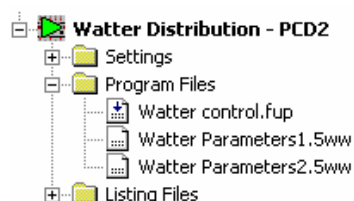
La *Finestra di Monitoraggio* viene visualizzata selezionando il comando di menu *View, Watch Window (Visualizza, Finestra di Monitoraggio)* oppure il pulsante di attivazione dedicato.

E' inoltre possibile predisporre svariate *Finestre di Monitoraggio* diverse nella cartella *Program File* del Project Manager. Per aggiungere un nuovo *File Watch Window* (Finestra di Monitoraggio - \*.5ww) si può utilizzare il comando di menu *File, New (File, Nuovo)* oppure il pulsante *Nuovo File*.



N.B.: I file di tipo \*.5ww non vengono mai linkati ad un progetto (nessuna freccia nell'icona del file). Le informazioni contenute in questi file non vengono considerate nella costruzione del programma.

Per aprire un file \*.5ww, è sufficiente selezionarlo facendo doppio clic con il mouse, oppure contrassegnare il file e selezionarlo con il menu *File, Open (File, Apri)*.



### 8.3.2 Aggiunta di dati ad una “Finestra di Monitoraggio” (Watch Window)

Trascinare i simboli dal programma o dall'editore dei simboli all'interno della *Finestra di Monitoraggio (Watch Window)*.

1. Posizionare il cursore del mouse al centro dell'icona del simbolo e premere il tasto sinistro

2. Tenere premuto il tasto del mouse e trascinare il simbolo nella Watch Window (Finestra di Monitoraggio)

3. Simboli con i relativi commenti e stati/valori

4. Avvio/Arresto Monitoraggio

Finestra di Monitoraggio

Symbol	Address	Value	Modify Value	Symbol Comment
HMS	R 2113	103034		PCD Clock with current time
DailyTimer	O 32	1		Daily Timer
Ontime	R 2115	60000		Switch on time
OFFTIME	R 2114	19000		Switch off time

E' inoltre possibile modificare direttamente i simboli all'interno della finestra.

Definizione nuovo indirizzo

Symbol	Address	Value	Modify Value	Symbol Comment
DailyTimer	O 32	1		Daily Timer
ONTIME	R 2005	60000		Switch on time

Symbol	Address	Value	Modify Value	Symbol Comment
DailyTimer	O 32	1		Daily Timer
ONTIME	R 2004	60000		Switch on time

Symbol	Address	Value	Modify Value	Symbol Comment
DailyTimer	O 32	1		Daily Timer
ONTIME	R 2005	60000		Switch on time
OFFTIME	R 2004	19000		Switch off time

### 8.3.3 Visualizzazione in linea dei dati



Avvio/Arresto Monitoraggio

Il pulsante di *Avvio/Arresto Monitoraggio* permette di visualizzare i valori attualmente presenti nel PCD associati a ciascun simbolo riportato nella Finestra di Monitoraggio (Watch Window).

Accertarsi che nella barra di stato della Finestra di Monitoraggio sia indicata la modalità RUN. Se necessario, forzare il PCD in RUN o STOP usando il menu *Online (In Linea)*.

### 8.3.4 Modifica in linea dei dati

La colonna *Modify Value (Modifica Valore)* permette di specificare nuovi valori per i vari simboli. I nuovi valori possono poi essere trasferiti al PCD usando il pulsante *Download Values (Trasferisci Valori)*.

The image shows two screenshots of the 'Daily timer.5sww \* [Daily timer]' window. The top screenshot shows the 'Modify Value' column with values 102433, 1, 83000, and 180000. A callout bubble points to the '180000' value with the text '1. Definire i nuovi valori'. The bottom screenshot shows the same window with a download icon (a blue arrow pointing down) and a callout bubble pointing to the '180000' value with the text '3. I nuovi valori sono ora caricati nel PCD'. A second callout bubble points to the download icon with the text '2. Trasferire i valori'.

### 8.3.5 Formato di visualizzazione

Il formato di visualizzazione dei valori può essere selezionato in base alle esigenze.

**Esempio:** Visualizzazione registro R 2004 in esadecimale.

The diagram illustrates the steps to change the display format of register values. It starts with a table of register addresses and values:

Address	Value
O 32	0
R 2003	113245
R 2004	182000
R 2005	53000

Next, a context menu is shown with options: Cut (Ctrl+X), Copy (Ctrl+C), Paste (Ctrl+V), Insert Line (Ins), Delete Line (Del), Move Up (Ctrl+Up), Move Down (Ctrl+Down), Units, and Show. A red arrow points from the 'R 2004' value in the table to the 'Show' option in the menu.

The 'Show' menu is expanded to show format options: Decimal (Ctrl+D), Floating Point (Ctrl+F),  Hex (Ctrl+H), ASCII, and Binary. A red arrow points from the 'Hex' option to the next table.

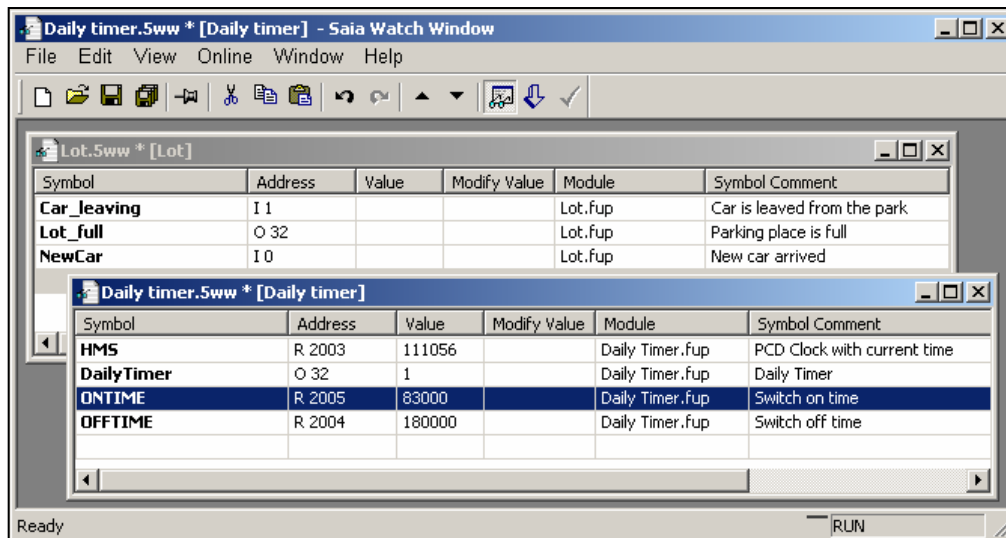
The final table shows the register values in hexadecimal format:

Address	Value
O 32	0
R 2003	113245
R 2004	0002C6F0H
R 2005	53000

### 8.3.6 “Finestra di Monitoraggio” (Watch Window) ed applicazioni multi-CPU

La *Finestra di Monitoraggio (Watch Window)* permette di aprire più documenti contemporaneamente. I menu, la barra degli strumenti e la barra di stato saranno in ogni caso sempre riferiti alla finestra attualmente attiva, ovvero quella identificata dalla barra di intestazione di colore blu.

Per default, ciascun documento aperto all'interno della Finestra di Monitoraggio adotta le *Impostazioni In Linea (Online settings)* associate alla CPU di appartenenza. E' quindi possibile visualizzare sulla rete di comunicazione i dati relativi ai vari PCD presenti nel progetto.

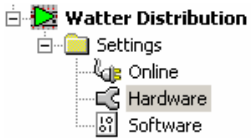


## 8.4 Configurazione

PG5 prevede due strumenti di configurazione:

- Il configuratore offline, a cui è possibile accedere dal menu *Hardware Settings (Impostazioni Hardware)* del Project Manager.
- Il configuratore online, a cui è possibile accedere con il menu Tools, *Online Configurator (Strumenti, Configuratore In Linea)*, oppure con il pulsante *Configuratore Online*.

### 8.4.1 Configuratore Offline (Fuori Linea)



Permette di configurare la memoria, i parametri di comunicazione e la password del PCD. Queste informazioni vengono salvate in un file di progetto PG5. L'utente deve utilizzare il pulsante *Trasferimento a PCD* per forzare il trasferimento dei dati di configurazione nella memoria del PCD.

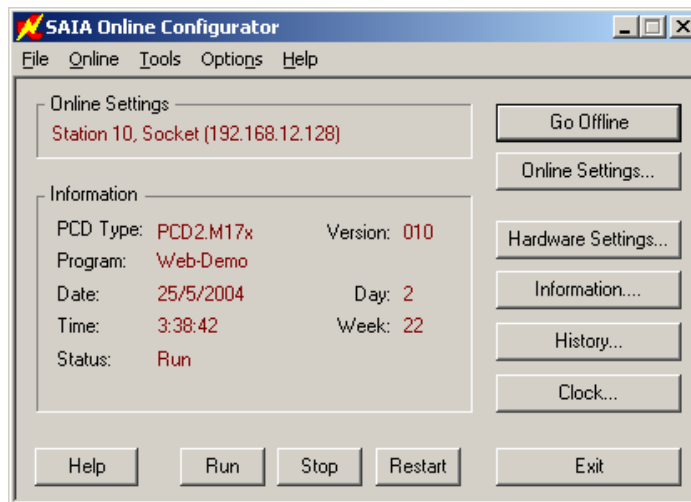
### 8.4.2 Configuratore Online (In Linea)



Configuratore  
Online

Permette di configurare la memoria, i parametri di comunicazione e la password del PCD. In questo caso, tuttavia, queste informazioni vengono scritte direttamente nella memoria del PCD. Nel progetto PG5 non rimarrà alcuna copia delle informazioni scritte. Senza un controllore, non rimane alcuna traccia di questa informazione. E' quindi consigliabile utilizzare il configuratore in linea per controllare i dati del PCD e configurarli con il comando di menu *Hardware Settings (Impostazioni Hardware)*.

### 8.4.3 Finestra del Configuratore Online



<i>PCD type</i>	Tipo di PCD
<i>Version</i>	Versione del firmware PCD
<i>Program Name</i>	Nome del programma utente
<i>Date</i>	Data dell'orologio PCD (nessun orologio: 1/1/92)
<i>Time</i>	Ora dell'orologio PCD
<i>Day</i>	Giorno della settimana: 1 = Lunedì, ... 7 = Sabato
<i>Week</i>	Numero di settimana
<i>Status</i>	Modo di funzionamento: Run, Stop, Halt, Conditional Run (Run Condizionato)
<i>Online Setting</i>	Collegamento diretto PGU o S-BUS



Se le informazioni indicate in rosso non vengono visualizzate, oppure viene visualizzato il messaggio *No response (Nessuna risposta)*, significa che non è possibile stabilire la comunicazione tra il PCD e il *Configuratore In Linea*.

In questo caso si prega di controllare i seguenti punti:

- Il computer è correttamente collegato al PCD con il cavo PCD8.K111/USB?
- I parametri di comunicazione sono stati selezionati correttamente con il pulsante *Settings (Impostazioni)*?

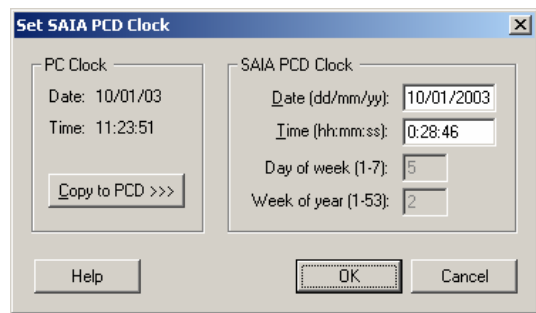


Configurazione della memoria PCD. Sono gli stessi parametri già descritti nelle *Impostazioni Hardware* del Project Manager.

### 8.4.4 Regolare l'orologio del PCD



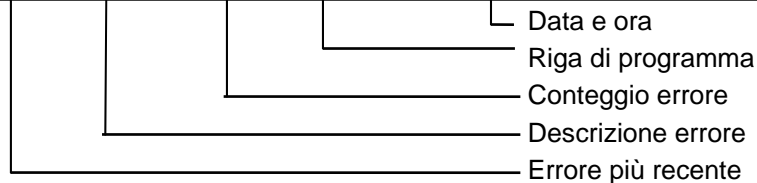
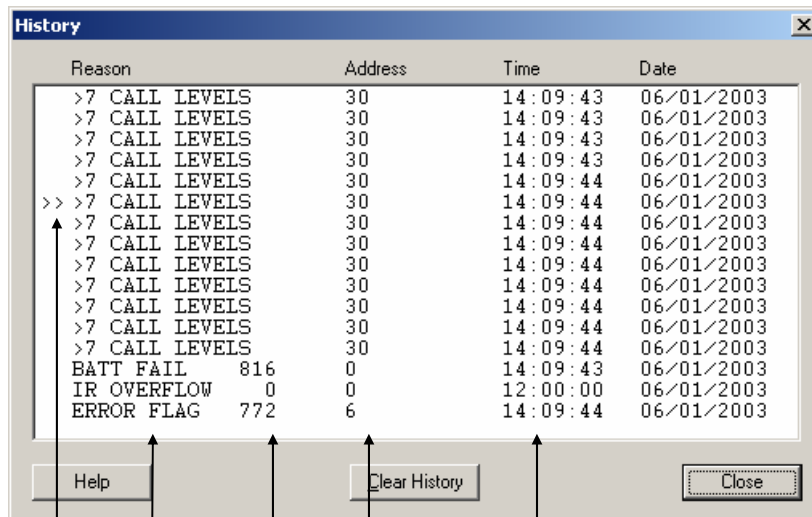
1. Selezionare il pulsante *Configuratore In Linea* nella finestra *SAIA Project Manager*. Quindi selezionare il pulsante *Clock (Orologio)*.
2. Copiare l'ora dal PC al controllore utilizzando il pulsante *Copy to PCD >>>* oppure, regolare l'orologio nei campi *SAIA PCD Clock (Orologio SAIA PCD)*.



### 8.4.5 Tabella Storica (History)



La tabella *History* registra tutti gli errori hardware o software che si verificano durante il funzionamento del PCD. Questa tabella viene aggiornata permanentemente, anche se non sono stati programmati i relativi blocchi XOB. Consultare la tabella "History" quando si accende il led di *Errore* della CPU.



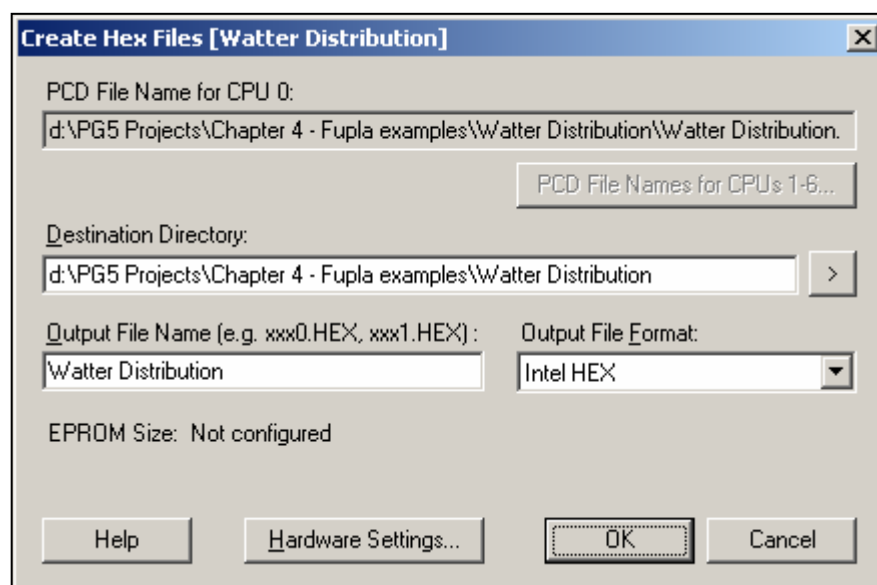
Note:

- Ogni CPU ha la propria tabella "History".
- L'errore *BATT FAIL (BATTERIA ESAURITA)* esiste solo sulla CPU 0.
- Se un errore può essere attribuito ad una riga di programma, questa verrà specificata. In caso contrario, l'errore verrà specificato in codice esadecimale.
- XOB 0 compare solo se è stato programmato.

## 8.5 Programmazione di memorie EPROM

PG5 supporta la creazione di file binari o esadecimali per tutti i tipi di programmatori EPROM standard presenti sul mercato.

Dall'ambiente Project Manager, selezionare il comando di menu *CPU, Create Hex Files ... (CPU, Creazione File Esadecimale ...)*.



Per creare un file EPROM:

- Configurare i parametri Hardware (Hardware Settings)
- Eseguire la costruzione del programma
- Selezionare il Formato del File di Output (Output File Format)
- Definire la Cartella di Destinazione (Destination Directory) e il Nome del File di Output (Output File Name)
- Premere il pulsante *OK*

### Dettagli tecnici

La EPROM contiene non solo il programma PCD, ma anche i parametri di configurazione hardware (*Hardware Settings*). Durante l'accensione, questi ultimi vengono automaticamente copiati dalla EPROM al PCD, ma solo se il PCD aveva perso le suddette informazioni (a causa di un problema di alimentazione dovuto ad un guasto od all'assenza della batteria).

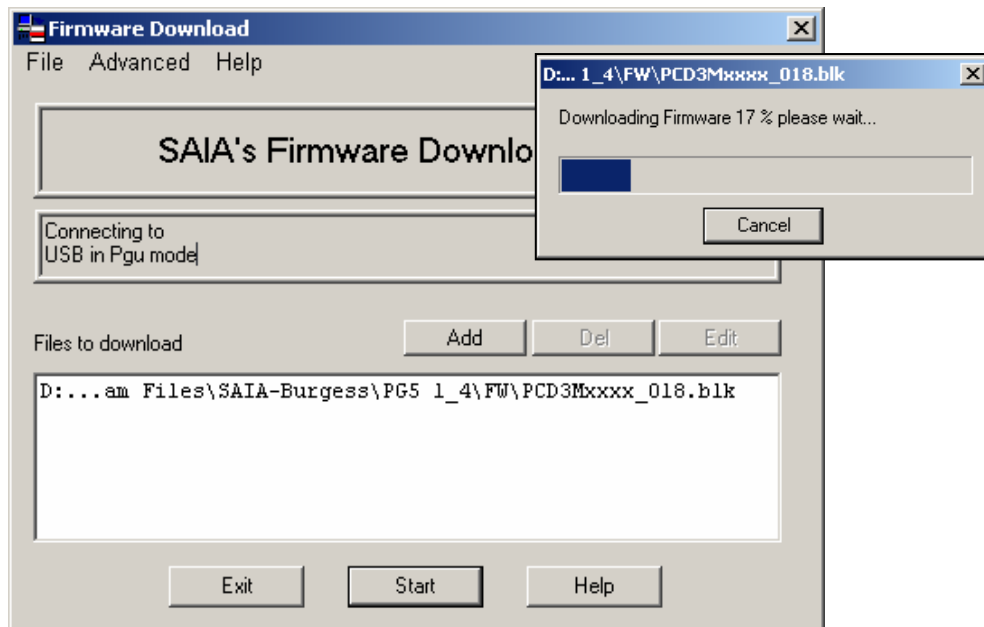
Quando si installa la EPROM, ricordarsi di posizionare correttamente i ponticelli della memoria nel PCD.

## 8.6 Aggiornamento del firmware (*Firmware Downloader*)

Talvolta è necessario ricorrere all'aggiornamento del firmware per poter usufruire delle ultime innovazioni di prodotto del PCD.

Per la maggior parte dei controllori, l'aggiornamento del firmware può essere effettuato solo sostituendo la EPROM.

Solo il firmware dei PCD più recenti<sup>1)</sup> può essere ricaricato in una memoria flash utilizzando un programma di utility a cui è possibile accedere dal menu *Tool, Firmware Downloader (Strumenti, Trasferimento Firmware)* del Project Manager.



### Istruzioni per il trasferimento del firmware:

- Il pulsante *ADD* aggiunge un nuovo file di firmware (\*.blk) all'elenco: *Files*.
- I file di firmware più recenti sono disponibili nella cartella *FW* sul CD di distribuzione di PG5.
- Utilizzare il comando di menu *File, Settings (File, Impostazioni)* per adattare i parametri di comunicazione alla modalità PGU (unica modalità attualmente supportata).
- Selezionare il firmware da trasferire al PCD.
- Collegare il cavo PCD8.K111 alla porta PGU del PCD.
- Spegnerne e successivamente accendere il PCD.
- Con PCD2.M480, premere due volte il pulsante *Run/Halt* mentre il LED *Run* sta ancora lampeggiando.
- Trasferire il firmware premendo il pulsante *Start*. Una finestra dialogo indica lo stato di avanzamento trasferimento dati.
- Completato il trasferimento dati, i LED *Run*, *Halt* e *Error* del PCD iniziano a lampeggiare. Questo significa che il PCD sta organizzando le informazioni nella propria memoria. Si prega quindi di attendere alcuni minuti prima di spegnere il controllore, o proseguire il lavoro.

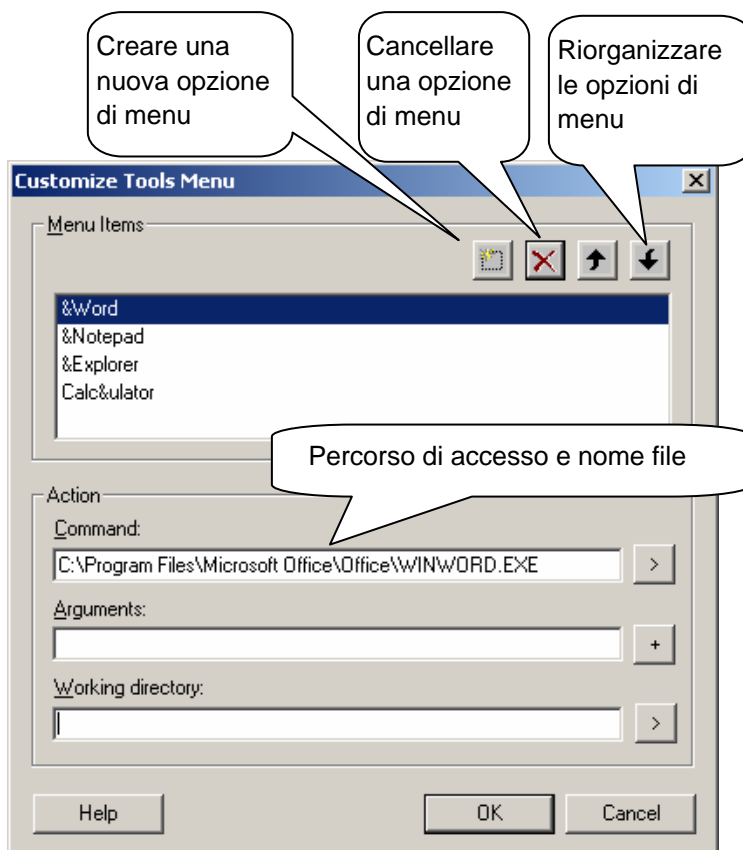
<sup>1)</sup> PCD2.M170, PCD4.M170, PCD2.M480

## 8.7 Menu utente

Il menu *Tools (Strumenti)* della finestra *SAIA Project Manager* può essere ampliato con l'aggiunta di scorciatoie che permettono di selezionare direttamente i vostri programmi preferiti.



Per aggiungere una scorciatoia, usare la voce di menu: *Tools, Customize Menu (Strumenti, Personalizza Menu)*. Usare la funzione di *Help* per maggiori dettagli.



## Indice

---

<b>9</b>	<b>Reti Saia (Saia Networks: S-Net)</b> .....	<b>2</b>
9.1	<i>Sommario</i> .....	2
9.2	<i>Scelta della rete</i> .....	2
9.2.1	Servizi supportati .....	2
9.2.2	Caratteristiche di progettazione .....	3

## 9 Reti Saia (Saia Networks: S-Net)

### 9.1 Sommario

Le soluzioni per l'automazione consistono spesso in vari controllori PCD decentralizzati, terminali e computer di supervisione collegati tra loro per mezzo di una rete di comunicazione. Ciascuna stazione controlla una parte del processo e provvede allo scambio dei dati con le restanti stazioni connesse alla rete.

Per garantire la necessaria flessibilità a tale concezione di controllo, il sistema PCD supporta diversi tipi di reti di comunicazione. Ciascuna rete possiede determinate caratteristiche e l'utente può quindi scegliere la rete più adatta alla propria applicazione.

Il pacchetto PG5 è uno strumento efficace per l'implementazione di queste soluzioni:

- Il *Project Manager Saia* fornisce una panoramica delle varie stazioni (PCD) e dei relativi parametri di configurazione, inclusi i parametri di comunicazione associati alla rete utilizzata.
- L'editore Fupla o IL permette di programmare opportunamente lo scambio dei dati tra le stazioni PCD connesse in rete.

Gli esempi di programmazione descritti nei seguenti capitoli sono tutti forniti ed installati assieme al pacchetto PG5 e servono come base di partenza per testare e comprendere la funzionalità di scambio dati attraverso vari tipi di reti PCD. Si noterà inoltre che molti esempi sono molto vicini a vere e proprie soluzioni pratiche.

### 9.2 Scelta della rete

La scelta del tipo di rete dipende dai requisiti richiesti dell'applicazione. Quelle seguenti sono le varie tipologie di reti SAIA (S-Net) disponibili:

- Profi-S-Bus : rete a bus di campo basata sullo standard Profibus FDL
- Ether-S-Bus : rete per lo scambio di informazioni basata sullo standard Ethernet
- Serial S-Bus : rete basata sull'interfaccia seriale RS 485/232
- S-Bus Modem : rete basata su una linea telefonica analogica o digitale
- Profi-S-IO : rete a bus di campo basata sullo standard Profibus DP
- Profibus DP: rete a bus di campo basata sullo standard Profibus DP

Le varie reti si distinguono per i servizi offerti, le caratteristiche tecniche e i relativi campi di applicazione.

#### 9.2.1 Servizi supportati

Nonostante tutte le reti supportino il trasporto degli elementi PCD quali ingressi, uscite, flag, registri, ecc.. alcune di queste supportano anche la programmazione, il controllo e la messa in servizio dei sistemi PCD via rete per mezzo del pacchetto PG5.

## 9.2.2 Caratteristiche di progettazione

### 9.2.2.1 Velocità di comunicazione

La velocità di comunicazione stabilisce il tempo di reazione per il trasferimento dei dati tra le stazioni. Se è necessario trasferire grandi quantità di dati oppure se è richiesto un tempo di reazione minimo, allora la velocità di comunicazione deve essere elevata. Notare che se la velocità di comunicazione via rete è personalizzabile, tutte le stazioni connesse alla rete devono operare alla stessa velocità.

### 9.2.2.2 Distanza massima

La distanza tra le stazioni può costituire una limitazione in caso di stazioni separate da grandi spazi. Non è possibile superare la distanza massima consentita senza provvedere all'amplificazione dei segnali elettrici per mezzo di un ripetitore di segnale o di uno switch / Hub. Generalmente, la distanza massima dipende anche dalla velocità di comunicazione adottata. Più la velocità è alta, minore sarà la distanza massima consentita. La riduzione della velocità di comunicazione può sovente rappresentare una soluzione valida per superare distanze maggiori.

### 9.2.2.3 Protocollo di comunicazione

Il "protocollo" è il formato del messaggio utilizzato per scambiare i dati tra le stazioni connesse alla rete. Il protocollo è paragonabile alla lingua usata tra due persone che parlano tra loro – esse si comprenderanno solo se entrambe parlano la stessa lingua. Analogamente, due stazioni possono scambiarsi dati solo se usano lo stesso protocollo di comunicazione.

I protocolli di alcune reti di comunicazione sono standard ufficiali. Ciò costituisce un notevole vantaggio quando apparecchi di costruttori diversi devono comunicare tra loro. Bus di campo e sensori usano sovente il protocollo Profibus DP.

Certe reti di comunicazione, quali Ethernet o Profibus FDL, supportano sulla stessa rete fisica lo scambio di dati usando protocolli diversi. In ogni caso, le due stazioni comunicanti devono utilizzare sempre lo stesso protocollo.

### 9.2.2.4 Scambio di dati in modalità "master-slave" o "multi-master"

Una rete "master-slave" è composta da una stazione "master" e da una serie di stazioni "slave". La stazione "master" controlla lo scambio dei dati tra le stazioni "slave".

Una rete "multi-master" è invece composta da più stazioni "master" e da una serie di stazioni "slave". Ciascuna stazione "master" può scambiare i dati con le altre stazioni "master" o "slave" connesse alla rete.

In entrambi i casi, tuttavia, non è consentito lo scambio di dati diretto tra due stazioni "slave".

### 9.2.2.5 Campi di applicazione

Alcune reti sono progettate per un utilizzo specifico. Ad esempio, il protocollo Profibus DP è orientato verso sistemi per il controllo di macchinari. Il protocollo di questa rete è ottimamente standardizzato e una vasta gamma di apparecchi compatibili forniti da diversi costruttori permettono lo scambio di dati sullo stesso bus usato per il comando di motori, ecc..

La rete Ether-S-Bus è invece più orientata verso sistemi di supervisione, server OPC oppure può essere semplicemente usata per la programmazione e la messa in servizio dei sistemi mediante il pacchetto PG5.

Il Serial S-Bus fornisce infine un modo molto semplice per collegare sistemi PCD. Si tratta di una rete molto economica che offre gli stessi servizi forniti da una rete Ether-S-Bus su una linea RS-485 ma anche attraverso linee telefoniche analogiche ed ISDN (Modem S-Bus).

## Reti di comunicazione S-Net

Servizi offerti :	Ether-S-Bus	Profi-S-Bus	Serial S-Bus	Modem S-Bus	Profi-S-IO Profibus DP
Programmazione PCD	Si	Si	Si	Si	No
Scambio di dati	Si	Si	Si	SI	Si
<b>Caratteristiche :</b>					
Velocità massima di trasmissione	10 e 100 Mbd	12 Mbd	38.4 /115.2 Kbd	38.4 /115.2 Kbd	12 Mbd
Distanza massima tra due stazioni senza ripetitore o switch/Hub	100 m	100 m	1200 m	-	100 m
Tipo di cavo	4 doppini intrecciati	1 doppino intrecciato	1 doppino intrecciato	-	1 doppino intrecciato
Protocollo	Saia	Saia	Saia	Saia	ISO normalizzato
Tipologia di scambio dei dati	Multi-Master	Multi-Master	Master-Slave	Multi-Master	Master-Slave
Numero massimo di stazioni collegabili	Illimitato	126	254	Illimitato	126
Campo di applicazione	Automazione industriale, building automation	Automazione industriale, building automation	Automazione industriale, building automation	Automazione industriale, building automation	Automazione industriale, building automation

La nuova rete Profi-S-Bus raggruppa tutti i vantaggi di una rete "multi-master" uniti ad un'elevata velocità di comunicazione all'interno di una rete a bus di campo ed è particolarmente adatta per applicazioni di automazione industriale.



## Indice

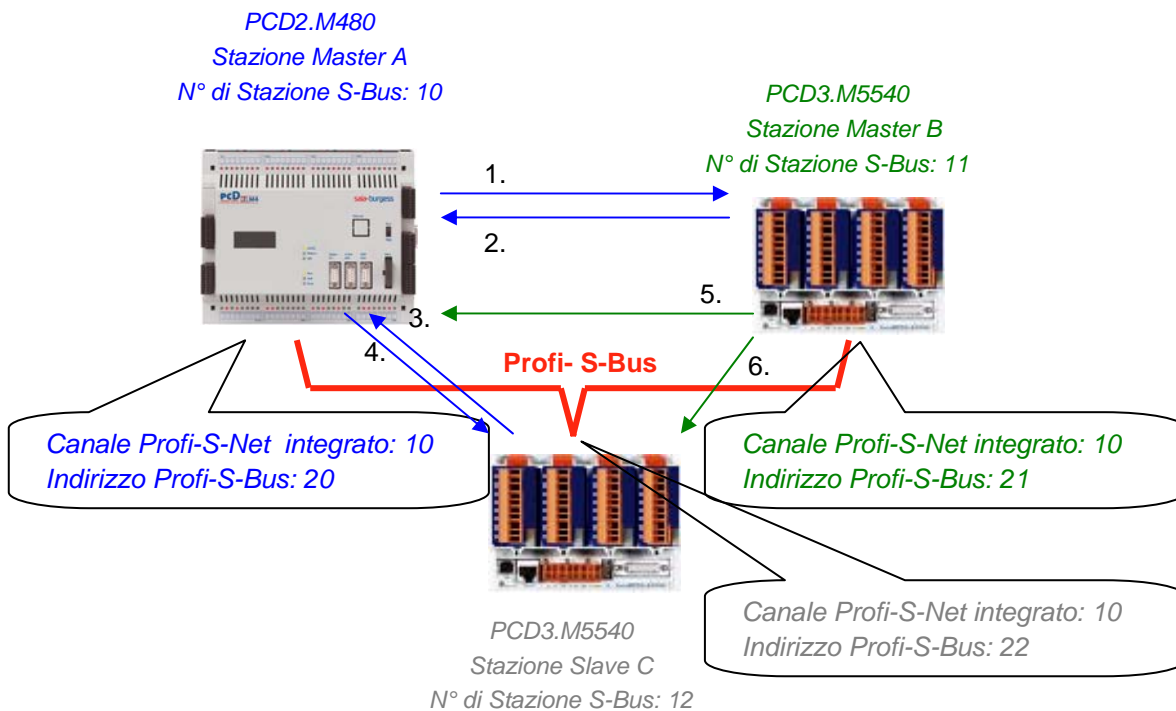
---

<b>10</b>	<b>Profi-S-Bus</b> .....	<b>2</b>
10.1	<i>Esempio di rete Profi-S-Bus</i> .....	2
10.2	<i>Esempi di Scambio Dati via Profi-S-Bus</i> .....	2
10.3	<i>Il Progetto PG5</i> .....	3
10.4	<i>Impostazioni Hardware delle stazioni Master e Slave</i> .....	3
10.4.1	Definizione dei parametri del PCD .....	3
10.4.2	Definizione del numero di stazione S-Bus all'interno della Rete .....	4
10.4.3	Definizione del canale di comunicazione Profi-S-Bus .....	4
10.4.4	Caricamento delle Impostazioni Hardware nella CPU .....	5
10.5	<i>Programma Fupla</i> .....	5
10.5.1	Assegnazione del canale mediante l'Fbox SASI .....	5
10.5.2	Assegnazione del canale Master .....	6
10.5.3	Assegnazione del canale Slave .....	6
10.5.4	Principi legati allo scambio dati in una rete "multi-master" .....	6
10.5.5	Scambio dati tra stazioni Master e Slave .....	7
10.5.6	Diagnostica .....	8
10.6	<i>Programma IL</i> .....	11
10.6.1	Assegnazione del Canale Master usando l'istruzione SASI .....	11
10.6.2	Assegnazione del Canale Slave .....	11
10.6.3	Principi legati allo scambio dati in una rete "multi-master" .....	11
10.6.4	Scambio dati tra stazioni Master e Slave .....	12
10.6.5	Diagnostica .....	13
10.7	<i>Funzione Gateway</i> .....	15
10.7.1	Applicazione .....	15
10.7.2	Configurazione della funzione Gateway PGU .....	16
10.7.3	Configurazione di una Porta Gateway Slave supplementare .....	18
10.7.4	Temporizzazioni di Comunicazione .....	19
10.8	<i>Ulteriori Informazioni/riferimenti</i> .....	20

## 10 Profi-S-Bus

Questo esempio illustra come ottenere lo scambio dati, quali Registri e Flag, tra stazioni PCD connesse ad una rete Profi-S-Bus.

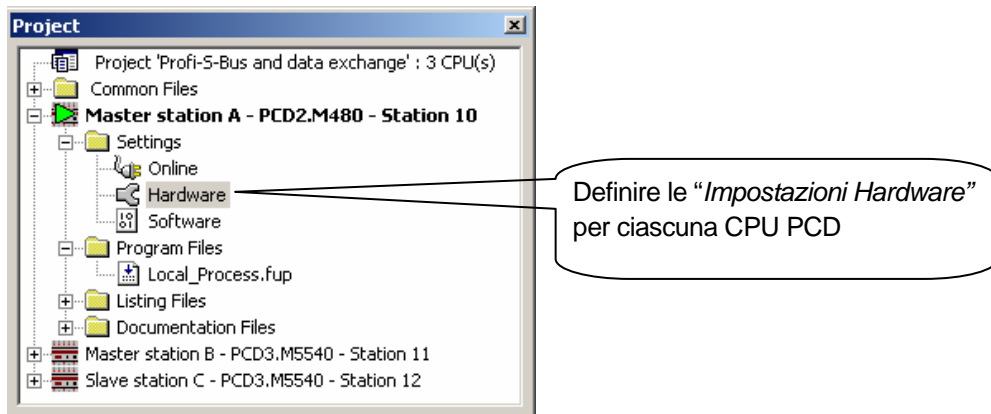
### 10.1 Esempio di rete Profi-S-Bus



### 10.2 Esempi di Scambio Dati via Profi-S-Bus

	<b>Master di controllo scambio dati</b>	<b>Dati sulla rete</b>	<b>Master passivo o Slave</b>
	<b>Stazione Master A</b>		<b>Stazione Master B</b>
1	Lampeggiatore0 .. 7 F 0 .. 7	Scrive 8 flag sulla stazione Master B	Stazione_A.Lampeggiatore0 .. 7 F 100 .. 107
2	Master_B.Valore100 R 125	Legge 1 registro dalla stazione Master B	Valore100 R 25
			<b>Stazione Slave C</b>
3	Slave_C.Binario0 .. 7 F 100 .. 107	Legge 8 flag dalla stazione Slave C	Binario0 .. 7 F 20 .. 27
4	Valore0 .. 5 R 0 .. 5	Scrive 6 registri sulla stazione Slave C	Master_A.Valore0 .. 5 R 20 .. 25
	<b>Stazione Master B</b>		<b>Stazione Master A</b>
5	Temperatura1 .. 4 Registri dinamici	Scrive le temperature rilevate sul Master A	Master_B.Temperatura1 .. 4 R 100 .. 104
			<b>Stazione Slave C</b>
6	Temperatura1 .. 4 Registri dinamici	Scrive le temperature rilevate sullo Slave C	Master_B.Temperatura1 .. 4 R 100 .. 104

## 10.3 Il Progetto PG5



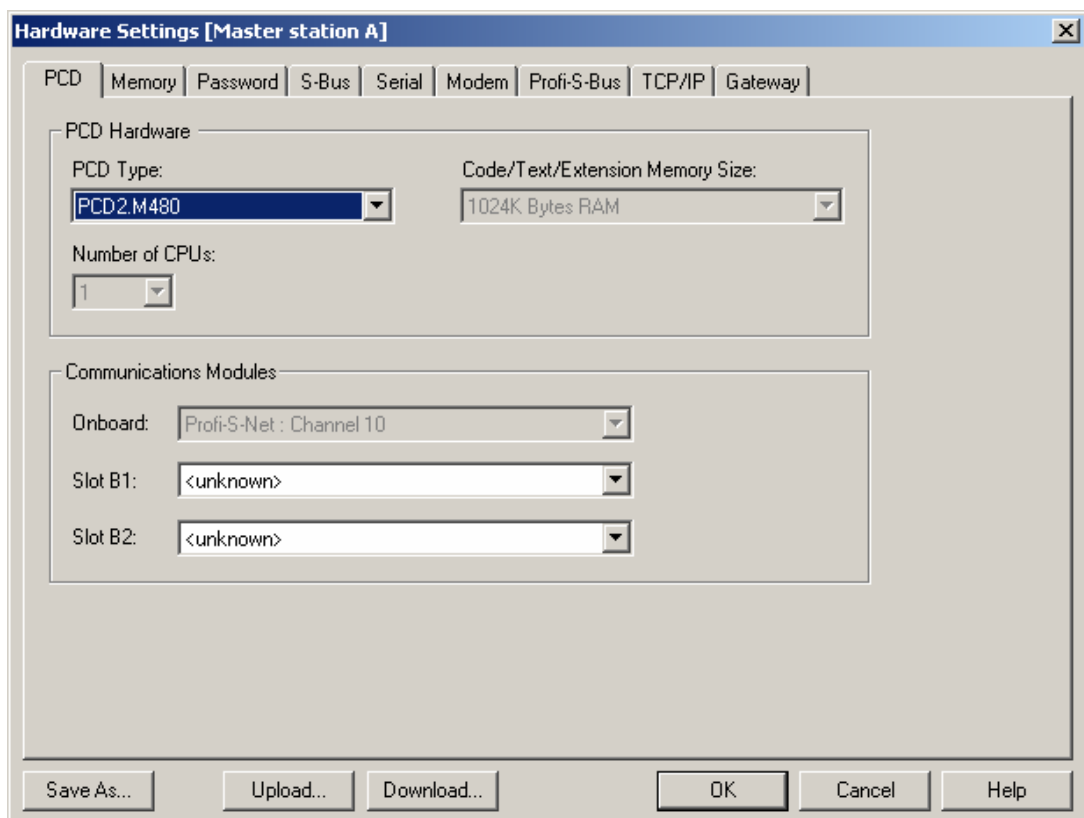
### *Project Manager Saia*

All'interno del *Project Manager Saia* sono definibili tutte le stazioni PCD utilizzate nel Progetto di un'applicazione nonché i parametri legati alla rete di comunicazione adottata. Si inizierà ora aggiungendo al Progetto una CPU per ciascuna delle Stazioni in Rete.

## 10.4 Impostazioni Hardware delle stazioni Master e Slave

La configurazione delle "Hardware Settings" (*Impostazioni Hardware*) per una stazione master e quella per una stazione slave sono simili.

### 10.4.1 Definizione dei parametri del PCD

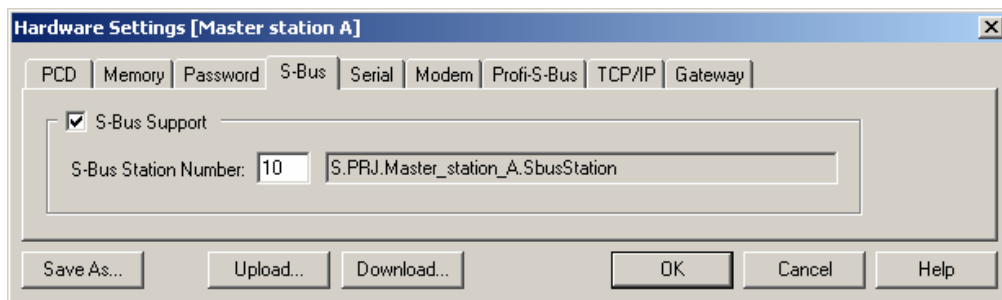


**PCD Type (Tipo di PCD)**

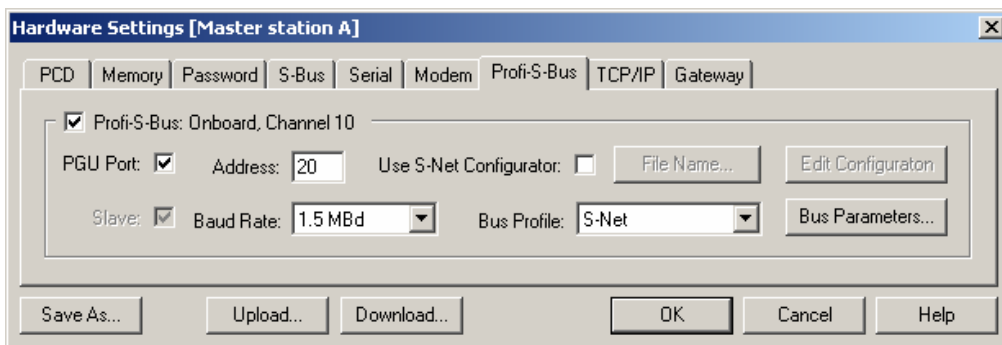
Permette di definire il tipo di CPU

**Communication Modules (Moduli di Comunicazione)**

Se necessario, permette di indicare il tipo di modulo di comunicazione inserito negli slot B1 e B2 del PCD2.M480.

**10.4.2 Definizione del numero di stazione S-Bus all'interno della Rete****S-Bus Station Number (Numero di Stazione S-Bus)**

Il numero di stazione S-Bus è comune per tutti i canali di comunicazione del PCD.

**10.4.3 Definizione del canale di comunicazione Profi-S-Bus****Address (Indirizzo)**

Numero di stazione Profi-S-Bus associato al canale.

**PGU Port o Slave (Porta PGU o Slave)**

Definisce il canale come Slave o PGU. Questa definizione può essere associata alla funzione master, aggiungendo un Fbox SASI nel programma Fupla.

**PGU Port (Porta PGU)**

Supporta lo scambio dati con stazioni master, sistemi di supervisione e terminali. Supporta anche la comunicazione con il pacchetto di programmazione PG5.

**Slave**

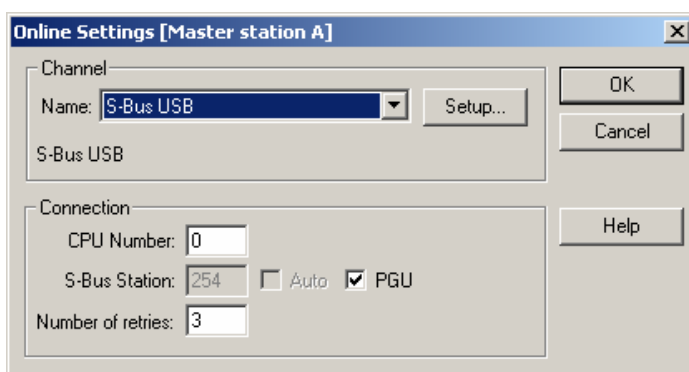
Supporta solo lo scambio dati con altre stazioni master, sistemi di supervisione e terminali.

**Baud Rate (Velocità di Comunicazione)**

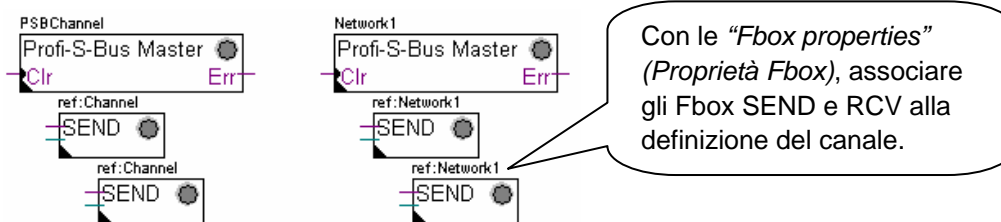
La velocità di comunicazione deve essere la stessa per tutte le stazioni connesse alla rete.

**S-Bus Profile (Profilo S-Bus)**

Le temporizzazioni per la trasmissione sono raggruppate in tre profili: S-Net, DP o user-defined (personalizzate). Selezionando quest'ultimo profilo, è possibile definire *temporizzazioni personalizzate* usando il pulsante "Bus Parameters..." (Parametri Bus). Il profilo selezionato deve essere identico per tutte le stazioni connesse alla rete. Quando all'interno della rete si usano dispositivi RIO PCD3.T76x, è necessario selezionare il profilo S-Net.

**10.4.4 Caricamento delle Impostazioni Hardware nella CPU**

Con i nuovi sistemi PCD2.M480 e PCD3, le *Impostazioni Hardware* possono essere caricate nella CPU per mezzo di un collegamento USB. E' solamente necessario definire il canale "S-Bus USB" all'interno della finestra "Online Settings" (Impostazioni Online). Caricare poi i parametri sul PCD usando il pulsante "Download" presente all'interno della finestra "Hardware Settings".

**10.5 Programma Fupla****10.5.1 Assegnazione del canale mediante l'Fbox SASI**

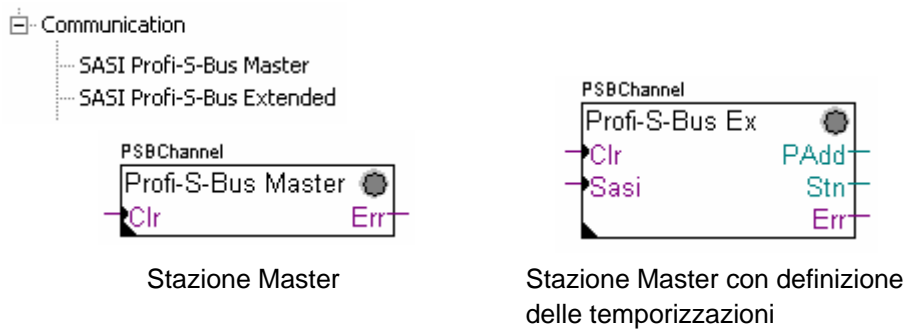
L'assegnazione si esegue usando un Fbox SASI posto all'inizio del file Fupla. Ciascuna rete di comunicazione richiede un proprio Fbox SASI, dal momento che i parametri differiscono in base al tipo di rete; lo stesso dicasi per le stazioni Master o Slave.

Se il PCD usa più canali di comunicazione, si deve definire ciascun canale usando il corrispondente Fbox SASI. Posizionare poi il cursore del mouse sull' Fbox SASI interessato e, per mezzo del menu contestuale, selezionare l'opzione "Fbox properties" quindi specificare un nome (Name) diverso per ogni canale. Questo nome permetterà di associare gli Fbox di scambio dati SEND e RCV all'Fbox SASI corrispondente al canale desiderato.

In base al tipo di rete selezionato, i parametri relativi al canale di comunicazione possono essere parzialmente definiti all'interno della finestra di configurazione (Adjust Window) dell'Fbox SASI. La definizione dei suddetti parametri può poi essere completata all'interno della finestra "Hardware Settings" (Impostazioni Hardware).

Il numero di Canale viene sempre definito all'interno della finestra di configurazione dell'Fbox SASI. Il numero di canale dipende dall'hardware del PCD e dall'hardware per la comunicazione utilizzato: slot B1, B2, interfaccia seriale PCD7.F, ...

### 10.5.2 Assegnazione del canale Master



L'assegnazione del canale Master è effettuabile combinando le Impostazioni Hardware con quelle associate a uno degli Fbox sopra indicati. Via Fbox sono definibili unicamente il canale di comunicazione e le temporizzazioni del Canale Master. Tutti gli altri parametri devono essere specificati mediante la finestra "Hardware Settings".

#### **Parametri della finestra di configurazione Fbox:**

##### **Channel (Canale)**

Definisce il canale di comunicazione associato all'interfaccia seriale collegata alla rete. Dipende dal tipo di PCD e dal relativo hardware.

##### **Timing (Temporizzazione)**

Il "Timeout" è tipicamente impostato al valore di default (0) e dovrà essere modificato solo per applicazioni speciali (Gateway).

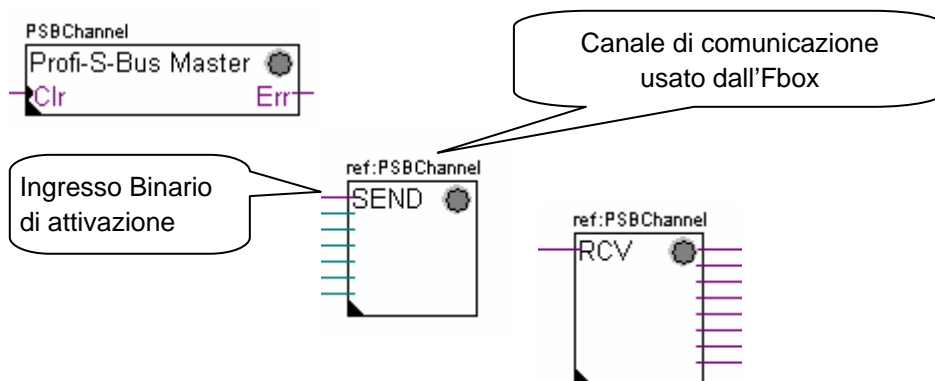
### 10.5.3 Assegnazione del canale Slave

Per il Canale Slave di una rete Profi-S-Bus non è necessario alcun Fbox SASI. Tutte le definizioni richieste sono già specificate all'interno della finestra "Hardware Settings".

### 10.5.4 Principi legati allo scambio dati in una rete "multi-master"

Una rete di comunicazione "multi-master" integra più di una stazione "master". Le Stazioni Master sono le uniche autorizzate a leggere o scrivere i dati da/su le restanti stazioni master e slave. Lo scambio dati tra stazioni slave non è consentito. Con una modalità di comunicazione "Multi-master", lo scambio dati avviene sempre tra le stazioni master connesse alla rete. Solo ad una delle stazioni master per volta viene assegnato un "token" che la autorizza a scambiare i dati con le restanti stazioni master o slave in rete. Quando il master in oggetto ha ultimato il trasferimento dei dati, il "token" viene passato alla stazione master successiva che a questo punto è autorizzata a scambiare i dati con le altre stazioni master o slave. Il "token" viene fatto circolare automaticamente tra le stazioni master, ma non viene mai assegnato alle stazioni slave. Pertanto, queste ultime non possono leggere o scrivere dati da/su altre stazioni della rete.

### 10.5.5 Scambio dati tra stazioni Master e Slave



Uno scambio dati tra stazioni, controllato dall'utente, è ottenibile inserendo all'interno delle pagine Fupla gli appositi Fbox Fupla selezionabili mediante l'“Fbox Selector” (*Selettore Fbox*). Sono disponibili Fbox per scrivere (SEND) o leggere (RCV) pacchetti di dati, in grado di supportare anche vari formati di dati: binario, intero, virgola mobile, Data Block, ecc..

Gli Fbox *SEND* o *RCV* possono essere ridimensionati al fine di aumentarne o diminuirne il numero di ingressi ed uscite, definendo così le dimensioni del pacchetto di dati che può essere scambiato con un'altra stazione.

L'indirizzo del Canale di Comunicazione usato dall'Fbox di trasmissione dati è indicato dal simbolo posto nella parte superiore sinistra dell'Fbox che lo collega all'Fbox SASI avente nome identico e in cui è stato definito l'indirizzo del canale. Questo simbolo può essere modificato posizionando il cursore del mouse sull'Fbox e selezionando l'opzione “Fbox Properties – Name” (*Proprietà Fbox – Nome*) all'interno del menu contestuale.

Ciascun Fbox *SEND* e *RCV* è dotato di un ingresso binario per l'attivazione dello scambio dati. Se questo ingresso viene mantenuto costantemente al livello alto, lo scambio dati verrà ripetuto il più velocemente possibile. Inviando invece un breve impulso a questo ingresso, lo scambio dati verrà eseguito almeno una volta, ma è sempre possibile forzarne l'esecuzione usando il pulsante “Execute” (Esegui) oppure mediante una Ripartenza a Freddo del PCD, comandata con l'opzione “Initialization” (*Inizializzazione*) presente all'interno della finestra di configurazione (*adjust window*).

I dati della stazione Master presenti sugli ingressi dell'Fbox *SEND* sono inviati alla stazione Slave definita nella finestra di configurazione. Mentre i dati presenti sulle uscite dell'Fbox *RCV* giungono dalla stazione slave definita dai parametri specificati all'interno della finestra di configurazione: indirizzo della stazione slave, elemento sorgente e indirizzo di base.

Solo per le stazioni Master è possibile programmare gli Fbox *SEND* e *RCV*! Alle stazioni slave è consentito solo assegnare il canale di comunicazione.

In base agli Fbox usati, la *finestra di configurazione* permette di definire le stazioni slave alle quali la stazione master può inviare (*SEND*) i dati oppure le stazioni slave dalle quali la stazione Master può leggere (*RCV*) i dati.

#### **Parametri della Finestra di Configurazione**

##### ***Profi-S-Bus Address (Indirizzo Profi-S-Bus)***

Definisce il numero della stazione slave Profi-S-Bus.

##### ***Source, destination station (Stazione Sorgente, Destinataria)***

Definisce il numero della stazione slave S-Bus.

**Source, destination element (Elemento Sorgente, Destinatario)**

Definisce il tipo di dati da scrivere o leggere sulla/dalla stazione slave.

**Source, destination address (Indirizzo Sorgente, Destinatario)**

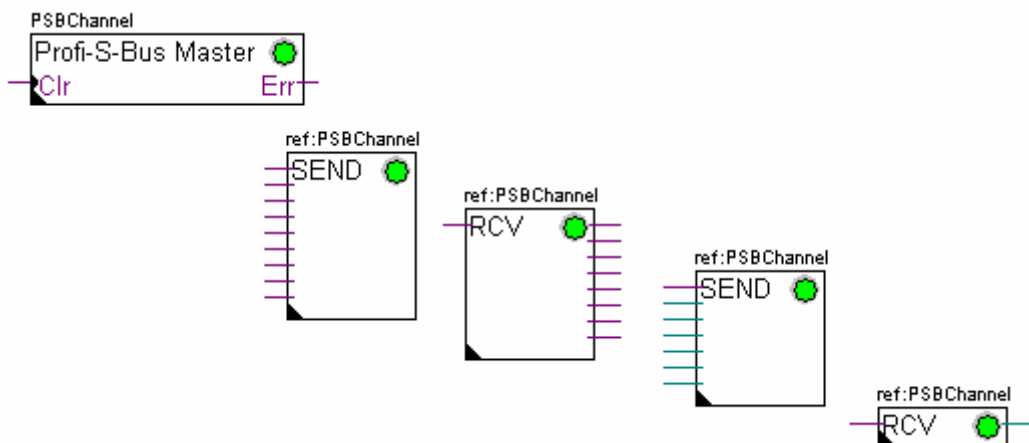
Definisce l'indirizzo di partenza del pacchetto di dati da scrivere o leggere sulla/dalla stazione slave. Il numero di valori di dati scambiati dipende dal numero di ingressi o di uscite dell'Fbox SEND o RCV.

**10.5.6 Diagnostica**

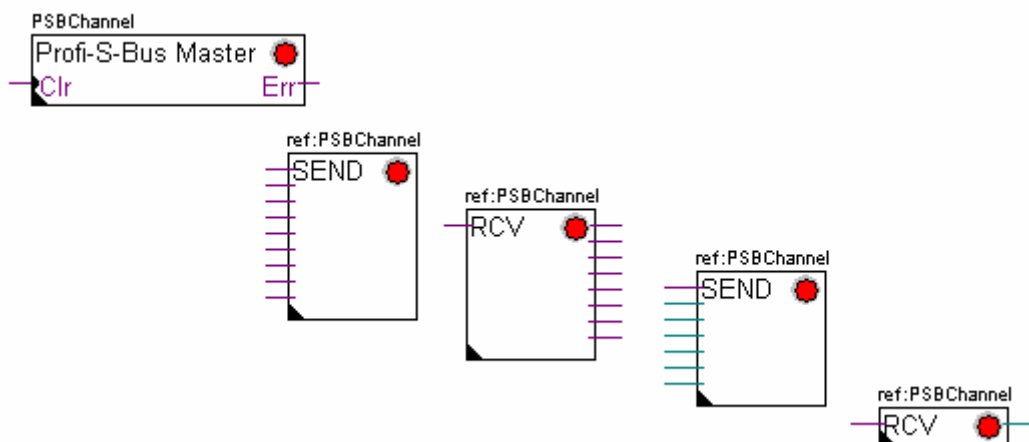
Se il programma è "Online", nell'angolo superiore destro degli Fbox *SASI*, *SEND* o *RCV* viene visualizzato un LED di colore verde o rosso. Se questo LED è di colore verde, significa che la trasmissione dati viene eseguita correttamente (OK); se di colore rosso indica invece una condizione di errore.

**Funzionamento corretto**

I LED di tutti gli Fbox sono di colore verde; lo scambio dati avviene correttamente.

**Nessun dato può essere scambiato all'interno della rete**

I LED dell'Fbox *SASI* e di tutti gli Fbox *SEND* e *RCV* sono di colore rosso; non è possibile trasferire alcun dato all'interno della rete.



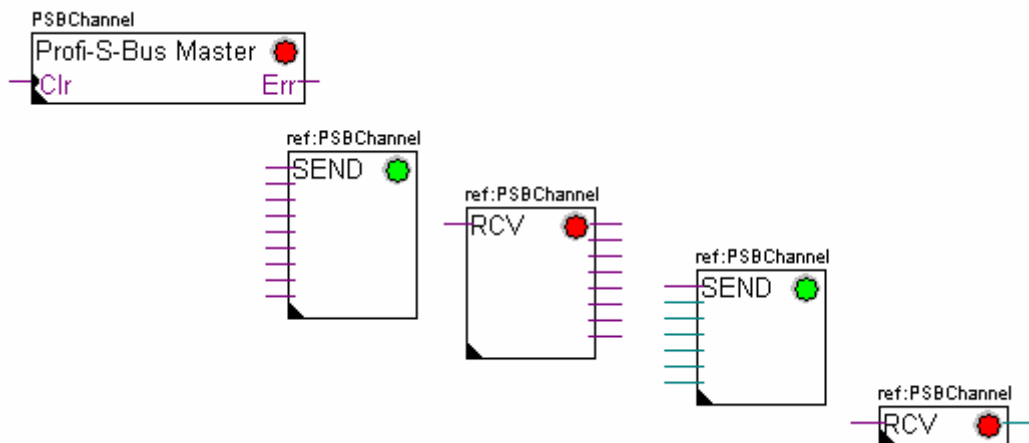


### Possibili azioni correttive eseguibili sulle stazioni master o slave:

- Verificare le impostazioni hardware all'interno della finestra "Hardware Settings"
- Verificare che le impostazioni hardware siano state correttamente caricate nel PCD
- Verificare che tutte le stazioni adottino lo stesso profilo: S-Net, DP
- Verificare che tutte le stazioni comunichino alla stessa velocità
- Verificare che il canale di comunicazione definito con la finestra "Hardware Settings" sia identico a quello definito con la funzione SASI (stesso numero di canale)
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Profi-S-Bus

### Solo alcuni Fbox non effettuano lo scambio dati

I LED dell'Fbox SASI e di alcuni Fbox SEND e RCV sono di colore rosso; gli Fbox il cui LED è di colore verde eseguono correttamente lo scambio dati



### Possibili azioni correttive eseguibili sulla stazione master

Verificare i parametri definiti all'interno della *finestra di configurazione* degli Fbox SEND e RCV il cui LED è di colore rosso.

Verificare che l'indirizzo della stazione slave specificato sia presente all'interno della rete.

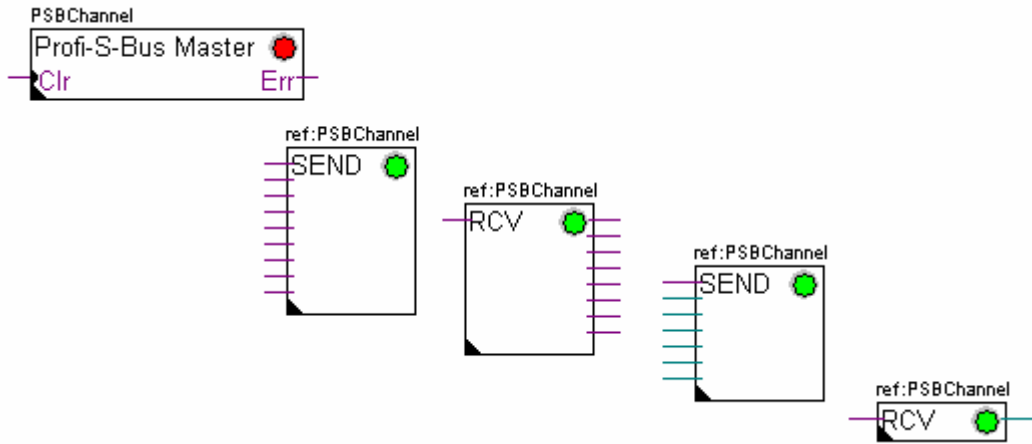
### Possibili azioni correttive eseguibili sulla stazione slave

Per tutti gli Fbox SEND e RCV il cui LED è di colore rosso, rilevare il numero di stazione slave specificato e verificare le corrispondenti stazioni.

- Verificare le impostazioni hardware all'interno della finestra "Hardware Settings"
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Profi-S-Bus

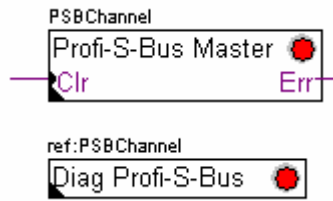
**Solo il LED dell’Fbox SASI è di colore rosso**

Aprire la finestra di configurazione dell’Fbox *SASI* ed eliminare l’ultimo allarme per mezzo del pulsante “Clear”.



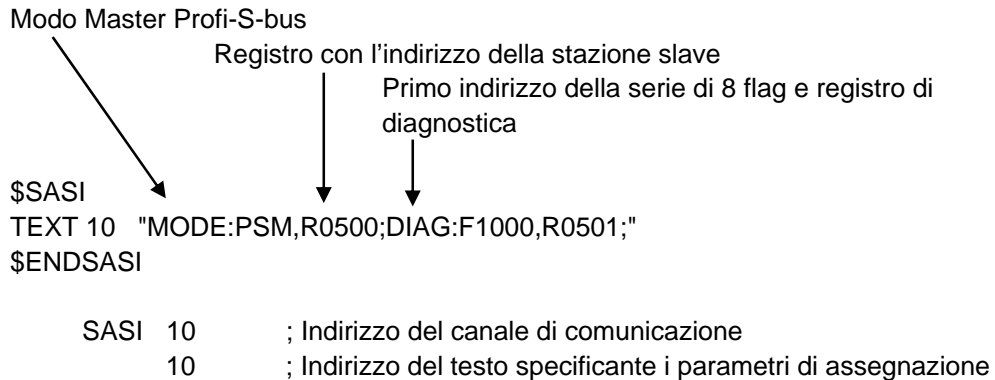
**Fbox di Diagnostica**

Quando il LED dell’Fbox SASI è di colore rosso, è sempre possibile ottenere una diagnosi consultando la finestra di configurazione della funzione *Diagnostica SASI*. Il corrispondente Fbox deve essere inserito immediatamente sotto l’Fbox *SASI*.



## 10.6 Programma IL

### 10.6.1 Assegnazione del Canale Master usando l'istruzione SASI



L'assegnazione del canale viene effettuata usando l'istruzione SASI posta all'inizio del programma: nella sequenza di inizializzazione Graftec o nel blocco di inizializzazione XOB 16.

L'istruzione SASI contiene due parametri: l'indirizzo del canale di comunicazione e l'indirizzo del testo che definisce tutti i parametri del canale.

I parametri di assegnazione specificati via Testo sono diversi da una rete di comunicazione all'altra così come per le diverse stazioni slave o master.

Se il PCD usa più canali di comunicazione, è necessario definire ciascun canale usando un'istruzione SASI ed un Testo di assegnazione.

In base al tipo di rete adottato, la definizione dei parametri del canale può essere completata con la finestra *"Hardware Settings"* (*Impostazioni Hardware*).

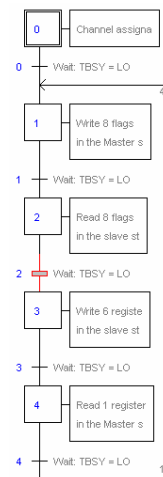
### 10.6.2 Assegnazione del Canale Slave

Per il Canale Slave di una rete Profi-S-Bus non è necessario definire alcuna istruzione SASI. Tutte le definizioni richieste sono già specificate all'interno della finestra *"Hardware Settings"*.

### 10.6.3 Principi legati allo scambio dati in una rete "multi-master"

Una rete di comunicazione "multi-master" integra più di una stazione "master". Le Stazioni Master sono le uniche autorizzate a leggere o scrivere i dati da/su le restanti stazioni master e slave. Lo scambio dati tra stazioni slave non è consentito. Con una modalità di comunicazione "Multi-master", lo scambio dati avviene sempre tra le stazioni master connesse alla rete. Solo ad una delle stazioni master per volta viene assegnato un "token" che la autorizza a scambiare i dati con le restanti stazioni master o slave in rete. Quando il master in oggetto ha ultimato il trasferimento dei dati, il "token" viene passato alla stazione master successiva che a questo punto è autorizzata a scambiare i dati con le altre stazioni master o slave. Il "token" viene fatto circolare automaticamente tra le stazioni master, ma non viene mai assegnato alle stazioni slave. Pertanto, queste ultime non possono leggere o scrivere dati da/su altre stazioni della rete.

### 10.6.4 Scambio dati tra stazioni Master e Slave



Passo Iniziale: assegnazione del canale

Passo: scambio dati

Transizione: attesa della fine dello scambio dati

Lo scambio dati tra le stazioni è un programma sequenziale: l'assegnazione del canale di comunicazione viene eseguita una sola volta e lo scambio dati all'interno della rete verrà effettuato solo se la precedente sessione di scambio dati è terminata. Per tali motivi si propone di definire la procedura di scambio dati in IL con l'Editore Graftec.

Il "Passo Iniziale" permette l'assegnazione del canale di comunicazione al Riavvio a Freddo del PCD.

I "Passi" successivi vengono eseguiti ciclicamente (loop) e un "passo" supporta lo scambio di un pacchetto di dati.

Ogni "passo" è separato da una "Transizione" che verifica lo stato del flag di diagnostica TBSY e stabilisce quindi se lo scambio dati è terminato. Si sarà autorizzati allo scambio dati definito dal "passo" seguente solo se il flag TBSY è allo stato "Basso".

#### **Scambio dati per mezzo di un Passo**

Prima di poter scambiare i dati, è necessario definire nel registro indirizzi dichiarato dal testo di assegnazione, l'indirizzo della stazione slave:

#### **Definizione dell'indirizzo della stazione slave**

LDL R 500 ; Registro indirizzi contenente l'indirizzo della stazione slave  
11 ; indirizzo S-Bus

LDH R 500 ; Registro indirizzi contenente l'indirizzo della stazione slave  
21 ; indirizzo Profi-S-Bus

Lo scambio dati tra le stazioni è ottenibile usando due istruzioni:

STXM: per la scrittura dei dati nella stazione slave (*SEND*)

SRXM: per la lettura dei dati dalla stazione slave (*RCV*)

Per ciascuna istruzione è necessario specificare quattro parametri: l'Indirizzo del Canale, il Numero di Dati da scambiare, l'Indirizzo di Base della Sorgente e Indirizzo di Base della Destinazione.

**Scrittura di 8 Flag (F 0... F 7) nella stazione slave (F 200... F 207)**

STXM 10 ; indirizzo del canale  
8 ; numero di dati da scambiare  
F 0 ; indirizzo di base della sorgente (stazione locale)  
F 200 ; indirizzo di base della destinazione (stazione slave)

**Lettura di un registro (in R 125) dalla stazione slave (R 25)**

SRXM 10 ; indirizzo del canale  
1 ; numero di dati da scambiare  
R 25 ; indirizzo di base della sorgente (stazione slave)  
R 125 ; indirizzo di base della destinazione (stazione locale)

Nota:

Solo le stazioni Master sono programmabili con le istruzioni STXM e SRXM ! Alle stazioni slave è consentito solo assegnare il canale di comunicazione.

**Attesa della “Fine Trasmissione” usando la Transizione**

STL F 1003 ; Verifica se TBSY si trova allo stato Basso

Il Testo di Assegnazione definisce una serie di 8 flag di diagnostica per la comunicazione. Il terzo di questi flag passa allo stato Alto durante la trasmissione dei dati ed allo stato Basso quando la sessione di scambio dati è terminata.

## 10.6.5 Diagnostica

**Assegnazione del canale**

In caso di problemi di comunicazione, verificare che l'assegnazione del canale venga eseguita correttamente. Analizzare il programma passo-passo e verificare che l'istruzione SASI non determini l'impostazione di un flag di errore. Se l'assegnazione del canale non viene eseguita in modo corretto, non sarà possibile stabilire la comunicazione.

**Possibili azioni correttive eseguibili sulle stazioni master o slave:**

- Verificare le impostazioni hardware all'interno della finestra “*Hardware Settings*”
- Verificare che le impostazioni hardware siano state correttamente caricate nel PCD
- Verificare che tutte le stazioni adottino lo stesso profilo: S-Net, DP
- Verificare che tutte le stazioni comunichino alla stessa velocità
- Verificare che il canale di comunicazione definito con la finestra “*Hardware Settings*” sia identico a quello definito con l'istruzione SASI (stesso numero di canale)
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Profi-S-Bus

### **I dati non vengono scambiati all'interno della rete**

Il Testo di Assegnazione definisce una serie di 8 flag di diagnostica per la comunicazione. Il quinto di questi flag (*TDIA: Diagnostica Trasmettitore*) passa allo stato Alto in caso di errore durante la trasmissione dati. Testare passo-passo il programma per rilevare le istruzioni STXM e SRXM errate.

Attenzione: in caso di errore di comunicazione, il flag di diagnostica TDIA rimane nello stato Alto finché non viene reimpostato a zero (resettato) il registro di diagnostica.

#### **Possibili azioni correttive eseguibili sulla stazione master**

Verificare i parametri delle istruzioni STXM e SRXM che generano l'errore. Verificare che l'indirizzo della stazione slave specificato sia presente all'interno della rete.

#### **Possibili azioni correttive eseguibili sulla stazione slave**

Per tutte le istruzioni STXM e SRXM che generano un errore, rilevare il numero di stazione slave specificato e verificare le corrispondenti stazioni.

- Verificare le impostazioni hardware all'interno della finestra "*Hardware Settings*"
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Profi-S-Bus

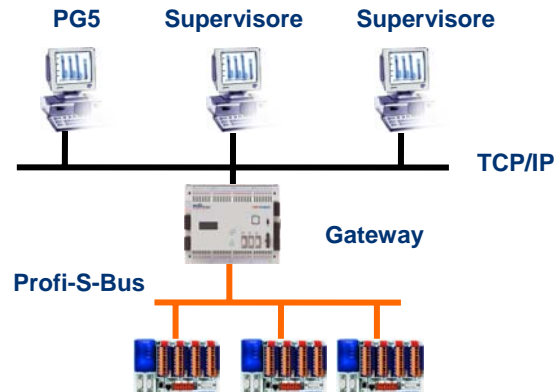
### **Registro di Diagnostica**

Il registro di diagnostica può fornire maggiori informazioni circa la natura dell'errore di comunicazione. Visualizzare il contenuto binario del registro e confrontarlo con le descrizioni riportate nel manuale del PCD o nel manuale dedicato alla Rete di Comunicazione utilizzata.

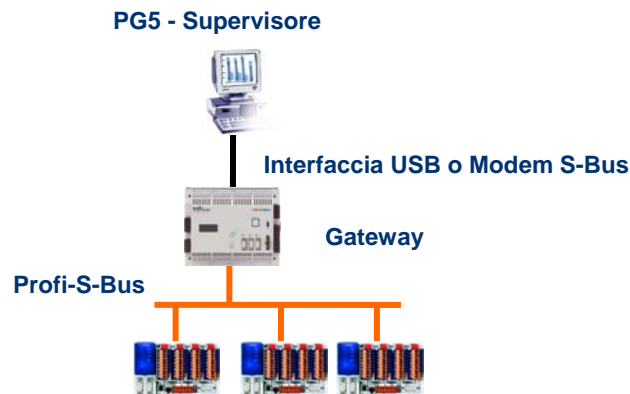
## 10.7 Funzione Gateway

La funzione *Gateway* viene tipicamente utilizzata per consentire a due reti di comunicazione diverse di comunicare tra loro oppure per usare il pacchetto di programmazione (PG5) o un sistema di supervisione (Visi+) con un tipo di rete diversa da quella normalmente supportata.

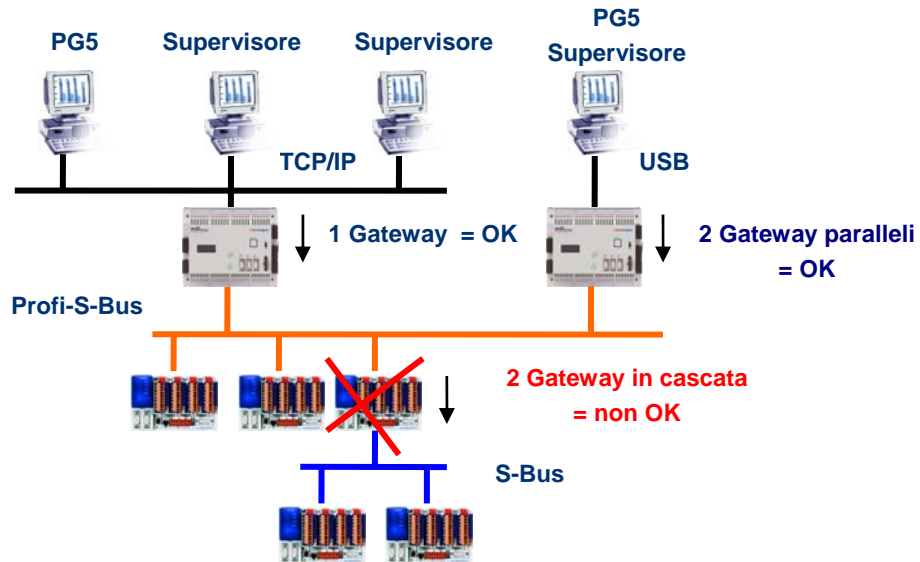
### 10.7.1 Applicazione



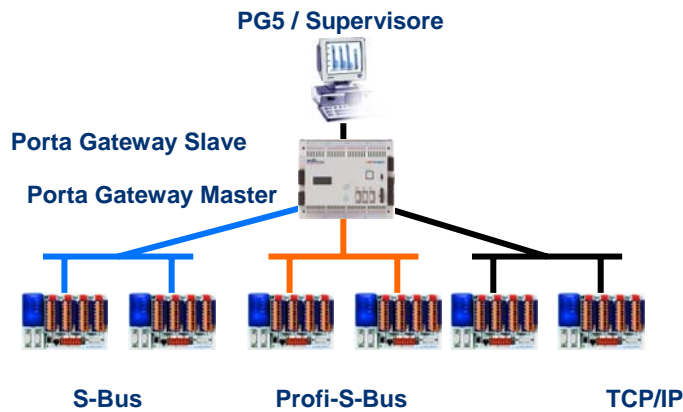
La funzione *Gateway* crea un "ponte" tra due reti consentendo, ad esempio, di collegare una rete Ethernet ad una rete Profi-S-Bus. In questo modo, i sistemi PCD possono scambiare dati attraverso un bus comune, specificatamente sviluppato per applicazioni di automazione, separato dalla rete informatica della società. Tuttavia, anche i PC su cui è installato il pacchetto PG5 o il sistema di supervisione Visi+ possono scambiare dati con i PCD.



La funzione *Gateway* può anche essere usata per creare un'interfaccia tra una rete di comunicazione ed il mondo esterno, realizzando, ad esempio, un'interfaccia di comunicazione USB o Modem.

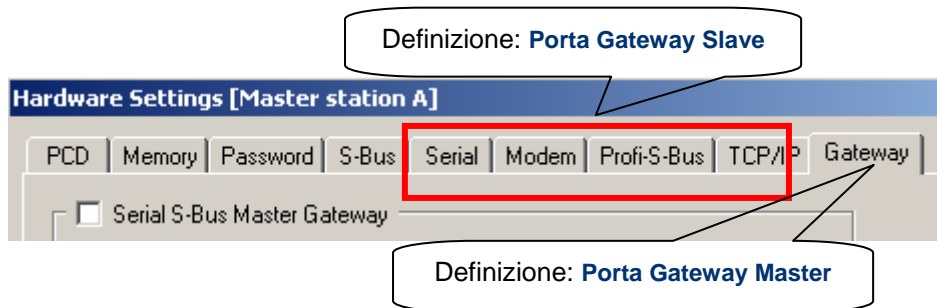


A causa della necessità di rispettare le temporizzazioni di comunicazione, non è possibile integrare due funzioni Gateway in cascata (serie). Tuttavia è possibile integrare due Gateway paralleli sulla stessa rete.



Quando necessario, un Gateway può essere usato per creare un “ponte” tra più sotto-reti di comunicazione.

### 10.7.2 Configurazione della funzione Gateway PGU



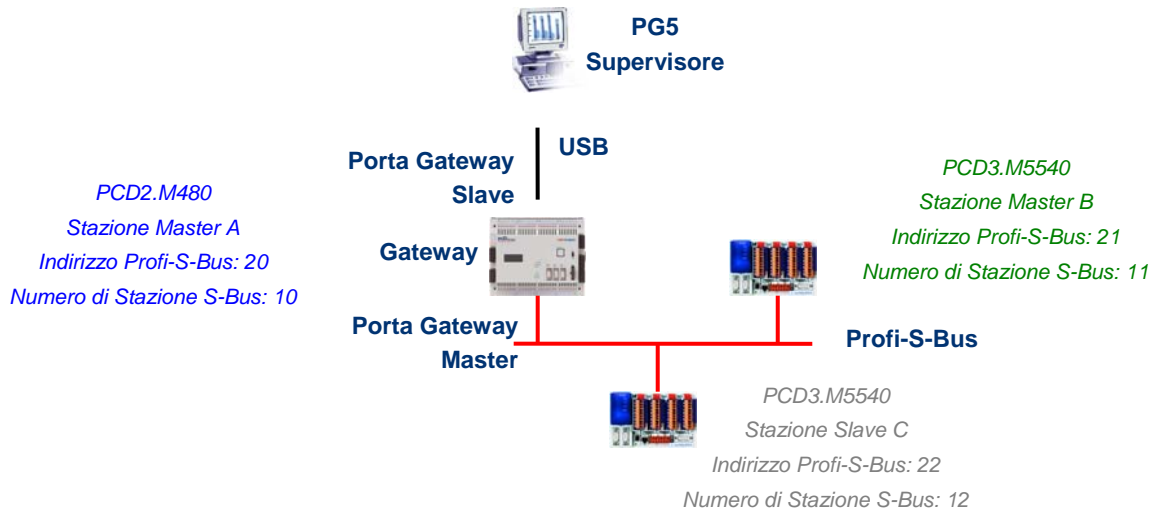
E' molto semplice configurare la funzione *Gateway*; non è infatti richiesta alcuna programmazione ma solo la definizione di alcuni parametri per il PCD all'interno della finestra “*Hardware Settings*” (*Impostazioni Hardware*).

Generalmente, è necessario definire solo una *Porta Gateway Slave* e una *Porta Gateway Master* per abilitare automaticamente la funzione *Gateway*.



Se il messaggio ricevuto dalla *Porta Gateway Slave* non è destinato alla stazione locale (*il Gateway*), allora i dati vengono ri-trasmessi ad una delle sotto-reti collegate alla *Porta Gateway Master*, in base al campo di indirizzi definito per la sotto-rete in oggetto.

### Esempio: Gateway USB, Profi-S-Bus



### Impostazioni Hardware per la Stazione Master A

Profi-S-Bus Master Gateway (Settings defined on Profi-S-Bus page)

Channel/Slot:

S-Bus Stations:  to

Il Gateway USB rappresenta un'eccezione, dato che non richiede la definizione di alcun parametro per la *Porta Gateway Slave* ma solo la definizione della *Porta Gateway Master* (non scordarsi di caricare la nuova configurazione sulla stazione Master A!).

### Impostazioni Online per le CPU usate nel progetto

Per realizzare una comunicazione via USB con ciascun PCD, all'interno della finestra *"Online Settings" (Impostazioni Online)* è necessario selezionare il canale USB e specificare il numero di stazione S-Bus.

## Verifica della funzionalità Gateway

### Slave station C - PCD3.M5540 - Station 12

Attivare una delle CPU *Master B* o *Slave C* del progetto e passare alla modalità "Online" per verificare la comunicazione con la stazione interessata.



Se necessario, l'"*Online Configurator*" (*Configuratore Online*) permette di verificare online il numero di stazione. E' anche possibile caricare il programma nella CPU attiva e, per verificarne il funzionamento, restare sempre collegati alla stazione *Master A* per mezzo di un cavo USB.

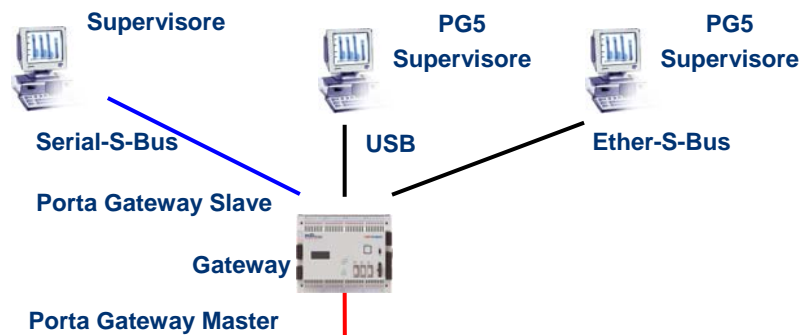
### Master station B - PCD3.M5540 - Station 11

Per comunicare con un'altra stazione della rete, attivare la relativa CPU e passare alla modalità "Online".

Nota:

Con la funzione *Gateway*, viene considerato il solo numero di stazione S-Bus; il numero di stazione Profi-S-Bus viene ignorato dal momento che i telegrammi vengono indirizzati a tutte le stazioni Profi-S-Bus (distribuzione Broadcast).

## 10.7.3 Configurazione di una Porta Gateway Slave supplementare



La Porta Gateway Slave rappresenta una via che permette di accedere alla rete dall'esterno. Se necessario, è possibile definire una seconda o una terza *Porta Gateway Slave*.

### Impostazioni Hardware

Generalmente, i PCD supportano solo un canale PGU slave. Tuttavia, i nuovi controllori PCD2.M480 e PCD3.Mxxxx supportano l'uso di più di una porta PGU sullo stesso PCD. La configurazione della seconda PGU Gateway Slave è effettuabile all'interno della finestra "*Hardware Settings*" (*Impostazioni Hardware*).

### Esempio: aggiunta di un Gateway supplementare Ether-S-Bus, Profi-S-Bus

The screenshot shows the **TCP/IP** configuration window for a Gateway Slave. The window title is "TCP/IP" and the selected tab is "TCP/IP : Slot B2, Channel 8". The configuration fields are as follows:
 

- TCP/IP**:  TCP/IP : Slot B2, Channel 8
- IP Node**: 13 (with a dropdown menu showing "S.PRJ.Master\_station\_A.IPNode")
- IP Address**: 192 . 168 . 12 . 130
- Subnet Mask**: 255 . 255 . 255 . 0
- Default Router**: 0 . 0 . 0 . 0
- PGU Port**:
- Slave**:
- Network Groups...**: A button to the right of the Slave checkbox.

La seconda *Porta PGU Gateway Slave* è stata aggiunta configurando il nodo e l'indirizzo TCP/IP all'interno della finestra "*Hardware Settings*" (*Impostazioni Hardware*). Se il controllore usato è un PCD2.M480, è necessario anche definire il modulo di comunicazione, selezionando per lo slot B2 del PCD il modulo PCD7.F65x (modulo Ethernet).

### Programmi Fupla o IL

Con i PCD di vecchia generazione ma anche con i nuovi PCD2.M480 e PCD3.Mxxxx, è possibile usare un Fbox/Istruzione SASI supplementare per aggiungere una *Porta Gateway Slave* supplementare. Questa *Porta Gateway Slave*, senza funzionalità PGU, non supporterà l'interazione con il pacchetto di programmazione PG5 ma solo con un terminale di supervisione. Inoltre, sarà ammessa solo la lettura/scrittura di dati PCD: registri, flag, ecc..

### Esempio Fupla: aggiunta di un Gateway supplementare Serial-S-Bus, Profi-S-Bus



E' necessario impostare a "Yes" il parametro di configurazione "Gateway". In base al tipo di canale, devono poi essere correttamente definiti i vari parametri presenti nella finestra di configurazione dell'Fbox.

### Esempio IL: aggiunta di un Gateway supplementare Serial-S-Bus, Profi-S-Bus

Usare il seguente Testo per assegnare il canale:

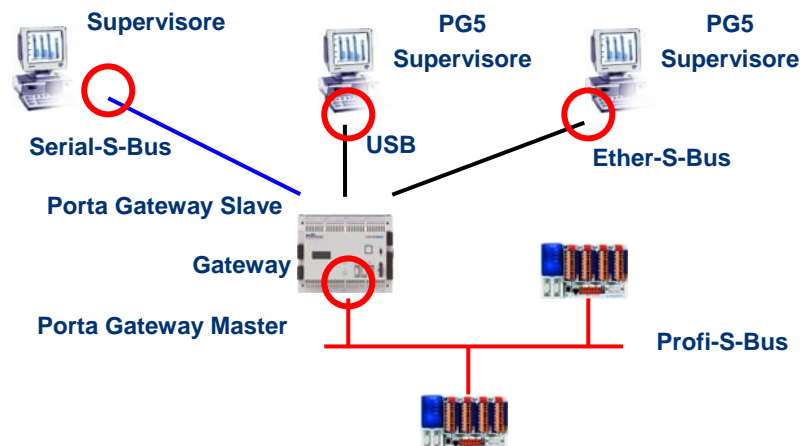
```
$SASI
```

```
TEXT 11 "UART:9600; MODE:GS2; DIAG:F1110, R0501;"
```

```
$ENDSASI
```

Flag e registro di diagnostica  
 Modo Gateway Slave S-Bus  
 Velocità di comunicazione

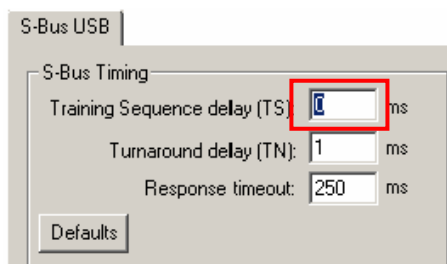
## 10.7.4 Temporizzazioni di Comunicazione



Generalmente, le *temporizzazioni* di comunicazione sono impostate ai valori di default e tale impostazione permette un funzionamento corretto del sistema. Tuttavia, implementando la funzione *Gateway* si aumentano i tempi di reazione richiesti per lo scambio dati.

Per tale motivo, a volte è necessario modificare l'intervallo di timeout delle stazioni master che usano la funzione *Gateway*. L'immagine precedente illustra quali sono i canali Master che richiedono una personalizzazione del timeout.

Per regolare il *Timeout* PG5, usare la finestra "*Online Settings*" (*Impostazioni Online*) associata alla *Stazione Master A*:



Per regolare il *Timeout* del programma di scambio dati del PCD, usare l'Fbox *SAS/Profi-S-Bus Extended*



## 10.8 Ulteriori Informazioni/riferimenti

Per maggiori dettagli, è possibile consultare i seguenti manuali/riferimenti:

- Manuale delle IstruzioniI 26/133
- Manuale Profi-S-Bus (in preparazione)
- Esempio di progetto Profi-S-Bus installato con il pacchetto PG5

## Indice

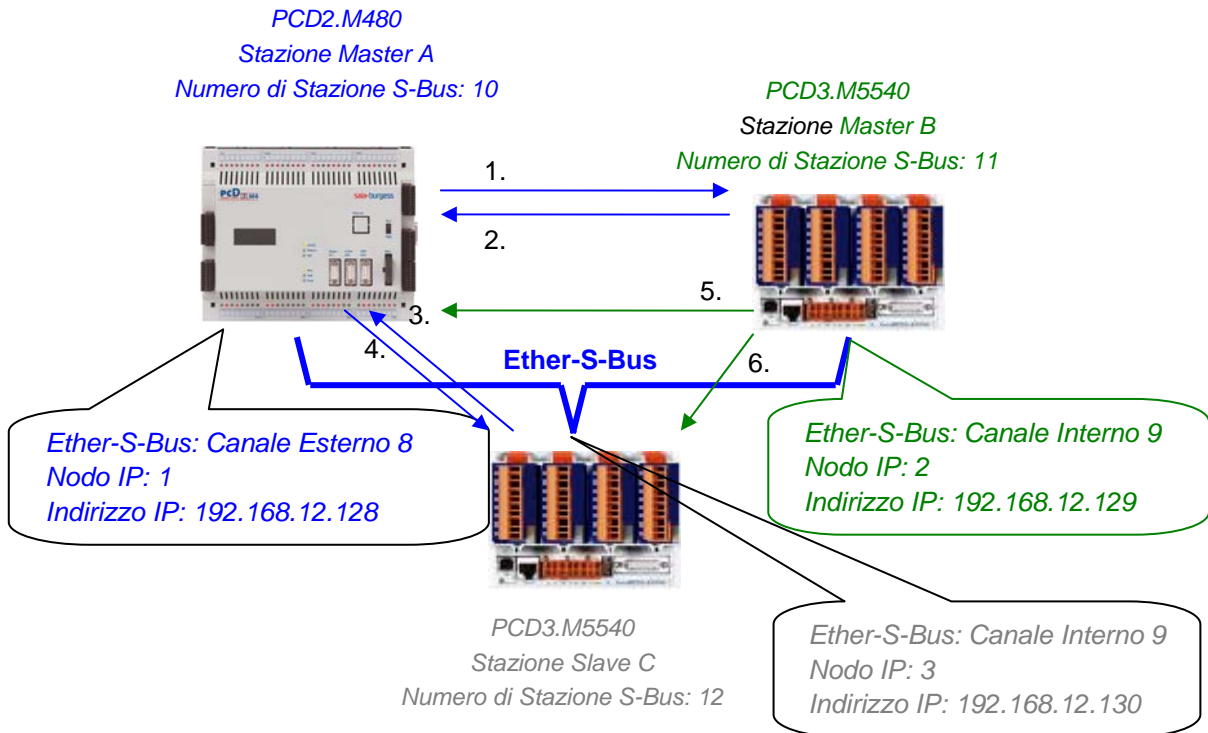
---

<b>11</b>	<b>Ether-S-Bus</b> .....	<b>2</b>
11.1	<i>Esempio di rete Ether-S-Bus</i> .....	2
11.2	<i>Esempi di Scambio Dati via Ether-S-Bus</i> .....	2
11.3	<i>Il Progetto PG5</i> .....	3
11.4	<i>Impostazioni Hardware delle stazioni Master e Slave</i> .....	3
11.4.1	Definizione dei parametri del PCD.....	3
11.4.2	Definizione del numero di stazione S-Bus all'interno della Rete .....	4
11.4.3	Definizione del canale di comunicazione Ether-S-Bus.....	4
11.4.4	Caricamento delle Impostazioni Hardware nella CPU .....	5
11.5	<i>Programma Fupla</i> .....	5
11.5.1	Assegnazione del canale mediante l'Fbox SASI .....	5
11.5.2	Assegnazione del canale Master .....	6
11.5.3	Assegnazione del Canale Slave .....	6
11.5.4	Principi legati allo scambio dati in una rete "multi-master" .....	6
11.5.5	Scambio dati tra stazioni Master e Slave.....	7
11.5.6	Diagnostica .....	8
11.6	<i>Programma IL</i> .....	11
11.6.1	Assegnazione del Canale Master usando l'istruzione SASI .....	11
11.6.2	Assegnazione del Canale Slave .....	11
11.6.3	Principi legati allo scambio dati in una rete "multi-master" .....	11
11.6.4	Scambio dati tra stazioni Master e Slave.....	12
11.6.5	Diagnostica .....	13
11.7	<i>Funzione Gateway</i> .....	15
11.7.1	Applicazione.....	15
11.7.2	Configurazione della funzione Gateway PGU.....	16
11.7.3	Configurazione di una Porta Gateway Slave supplementare .....	18
11.7.4	Temporizzazioni di Comunicazione .....	20
11.8	<i>Ulteriori Informazioni/riferimenti</i> .....	20

# 11 Ether-S-Bus

Questo esempio illustra come ottenere lo scambio dati, quali Registri e Flag, tra stazioni PCD connesse ad una rete Ether-S-Bus.

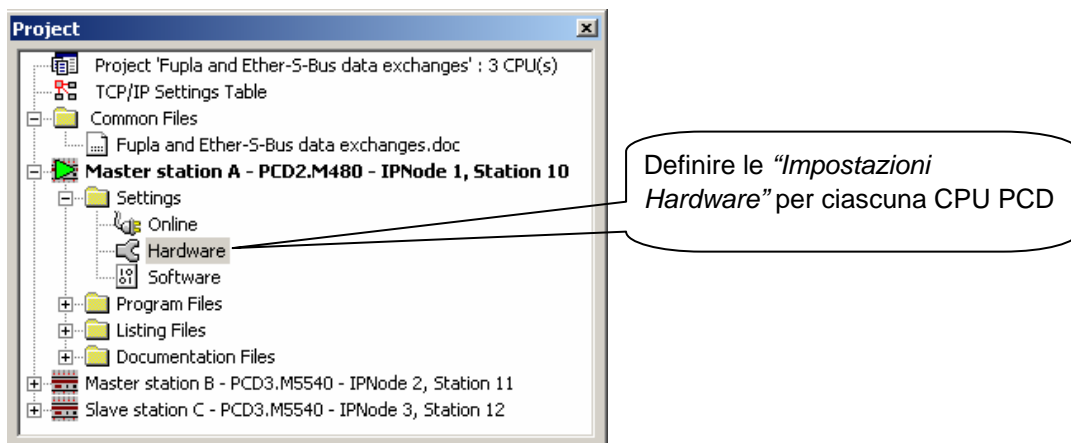
## 11.1 Esempio di rete Ether-S-Bus



## 11.2 Esempi di Scambio Dati via Ether-S-Bus

	<b>Master di controllo scambio dati</b>	<b>Dati sulla rete</b>	<b>Master passivo o Slave</b>
	<b>Stazione Master A</b>		<b>Stazione Master B</b>
1	Lampeggiatore0 .. 7 F 0 .. 7	Scrive 8 flag sulla stazione Master B	Stazione_A.Lampeggiatore0 .. 7 F 100 .. 107
2	Master_B.Valore100 R 125	Legge 1 registro dalla stazione Master B	Valore100 R 25
			<b>Stazione Slave C</b>
3	Slave_C.Binario0 .. 7 F 100 .. 107	Legge 8 flag dalla stazione Slave C	Binario0 .. 7 F 20 .. 27
4	Valore0 .. 5 R 0 .. 5	Scrive 6 registri sulla stazione Slave C	Master_A.Valore0 .. 5 R 20 .. 25
	<b>Stazione Master B</b>		<b>Stazione Master A</b>
5	Temperatura1 .. 4 Registri dinamici	Scrive le temperature rilevate sul Master A	Master_B.Temperatura1 .. 4 R 100 .. 104
			<b>Stazione Slave C</b>
6	Temperatura1 .. 4 Registri dinamici	Scrive le temperature rilevate sullo Slave C	Master_B.Temperature1 .. 4 R 100 .. 104

### 11.3 Il Progetto PG5



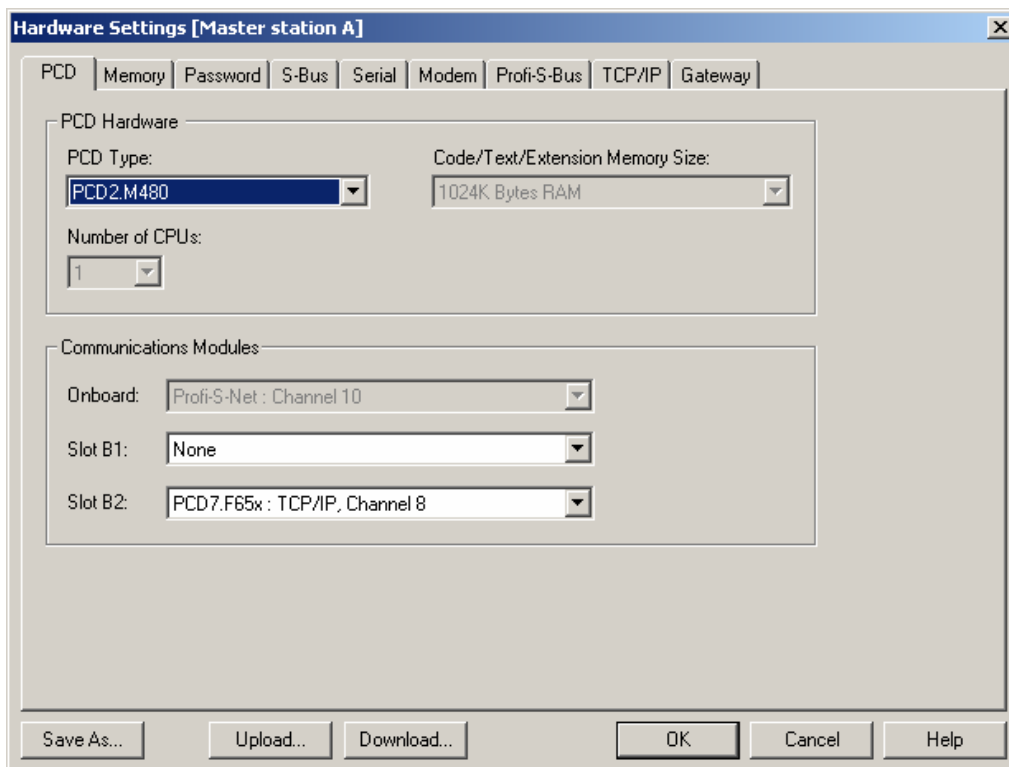
#### Project Manager Saia

All'interno del *Project Manager Saia* sono definibili tutte le stazioni PCD utilizzate nel Progetto di un'applicazione nonché i parametri legati alla rete di comunicazione adottata. Si inizierà ora aggiungendo al Progetto una CPU per ciascuna delle Stazioni in Rete.

### 11.4 Impostazioni Hardware delle stazioni Master e Slave

La configurazione delle "Hardware Settings" (*Impostazioni Hardware*) per una stazione master e quella per una stazione slave sono simili.

#### 11.4.1 Definizione dei parametri del PCD

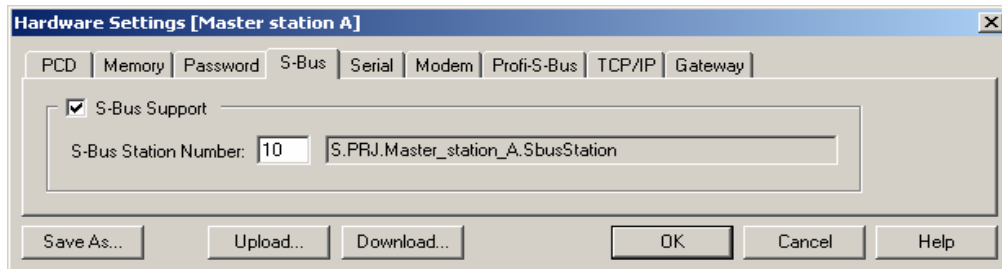


**PCD Type (Tipo di PCD)**

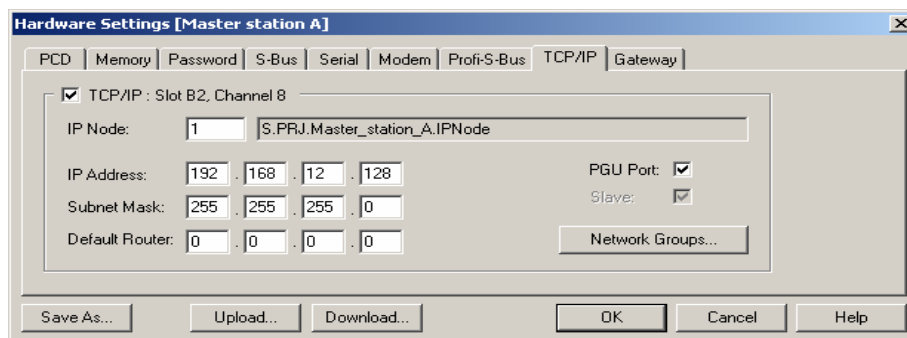
Permette di definire il tipo di CPU

**Communication Modules (Moduli di Comunicazione)**

Se necessario, permette di indicare il tipo di modulo di comunicazione inserito negli slot B1 e B2 del PCD2.M480.

**11.4.2 Definizione del numero di stazione S-Bus all'interno della Rete****S-Bus Station Number (Numero di Stazione S-Bus)**

Il numero di stazione S-Bus è comune per tutti i canali di comunicazione del PCD.

**11.4.3 Definizione del canale di comunicazione Ether-S-Bus****IP Address (Indirizzo IP)**

Numero di stazione Ether-S-Bus associato al canale.

**IP Node (Nodo IP)**

Numero del Nodo TCP/IP. Il nodo viene usato negli Fbox SEND e RCV per definire la stazione Slave con cui devono essere scambiati i dati.

**PGU Port o Slave (Porta PGU o Slave)**

Definisce il canale come Slave o PGU. Questa definizione può essere associata alla funzione master, aggiungendo un Fbox SASI nel programma Fupla.

**PGU Port (Porta PGU)**

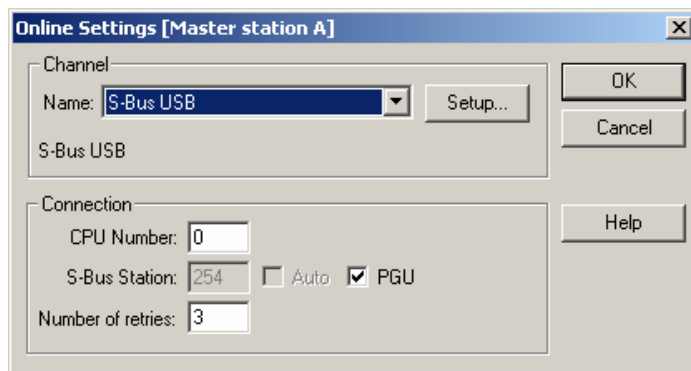
Supporta lo scambio dati con stazioni master, sistemi di supervisione e terminali. Supporta anche la comunicazione con il pacchetto di programmazione PG5.

**Slave**

Supporta solo lo scambio dati con altre stazioni master, sistemi di supervisione e terminali.



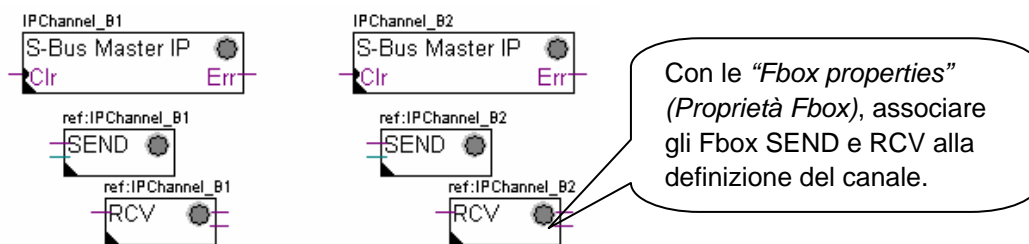
### 11.4.4 Caricamento delle Impostazioni Hardware nella CPU



Con i nuovi sistemi PCD2.M480 e PCD3, le *Impostazioni Hardware* possono essere caricate nella CPU per mezzo di un collegamento USB. E' solamente necessario definire il canale "S-Bus USB" all'interno della finestra "Online Settings" (Impostazioni Online). Caricare poi i parametri sul PCD usando il pulsante "Download" presente all'interno della finestra "Hardware Settings".

## 11.5 Programma Fupla

### 11.5.1 Assegnazione del canale mediante l'Fbox SASI



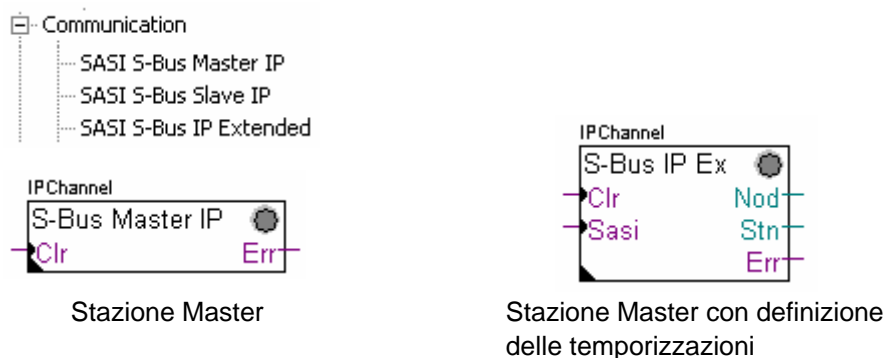
L'assegnazione si esegue usando un Fbox SASI posto all'inizio del file Fupla. Ciascuna rete di comunicazione richiede un proprio Fbox SASI, dal momento che i parametri differiscono in base al tipo di rete; lo stesso dicasi per le stazioni Master o Slave.

Se il PCD usa più canali di comunicazione, si deve definire ciascun canale usando il corrispondente Fbox SASI. Posizionare poi il cursore del mouse sull' Fbox SASI interessato e, per mezzo del menu contestuale, selezionare l'opzione "Fbox properties" quindi specificare un nome (Name) diverso per ogni canale. Questo nome permetterà di associare gli Fbox di scambio dati SEND e RCV all'Fbox SASI corrispondente al canale desiderato.

In base al tipo di rete selezionato, i parametri relativi al canale di comunicazione possono essere parzialmente definiti all'interno della finestra di configurazione (Adjust Window) dell'Fbox SASI. La definizione dei suddetti parametri può poi essere completata all'interno della finestra "Hardware Settings" (*Impostazioni Hardware*).

Il numero di Canale viene sempre definito all'interno della finestra di configurazione dell'Fbox SASI. Il numero di canale dipende dall'hardware del PCD e dall'hardware per la comunicazione utilizzato: slot B1, B2, interfaccia seriale PCD7.F, ...

## 11.5.2 Assegnazione del canale Master



L'assegnazione del canale Master è effettuabile combinando le Impostazioni Hardware con quelle associate a uno degli Fbox sopra indicati.

### **Parametri della finestra di configurazione Fbox:**

#### ***Channel (Canale)***

Definisce il numero del canale di comunicazione collegato alla rete. Dipende dal tipo di PCD e dal relativo hardware.

#### ***Timing (Temporizzazione)***

Il "Timeout" è tipicamente impostato al valore di default (0) e dovrà essere modificato solo per applicazioni speciali (Gateway).

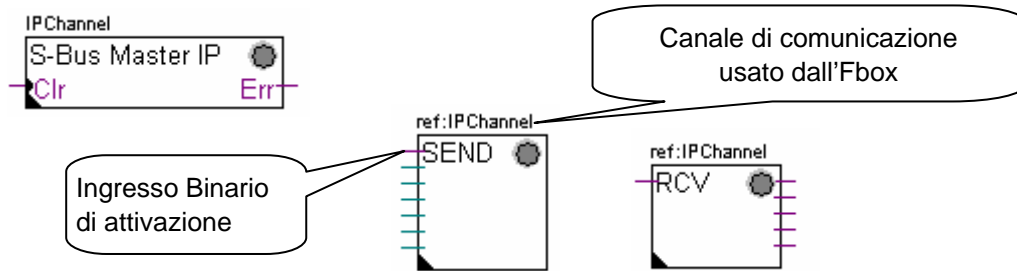
## 11.5.3 Assegnazione del Canale Slave

Per il Canale Slave di una rete Ether-S-Bus non è necessario alcun Fbox SASI. Tutte le definizioni richieste sono già specificate all'interno della finestra "Hardware Settings".

## 11.5.4 Principi legati allo scambio dati in una rete "multi-master"

Una rete di comunicazione "multi-master" integra più di una stazione "master". Le Stazioni Master sono le uniche autorizzate a leggere o scrivere i dati da/su le restanti stazioni master e slave. Lo scambio dati tra stazioni slave non è consentito. Con una modalità di comunicazione "Multi-master", lo scambio dati avviene sempre tra le stazioni master connesse alla rete. Solo ad una delle stazioni master per volta viene assegnato un "token" che la autorizza a scambiare i dati con le restanti stazioni master o slave in rete. Quando il master in oggetto ha ultimato il trasferimento dei dati, il "token" viene passato alla stazione master successiva che a questo punto è autorizzata a scambiare i dati con le altre stazioni master o slave. Il "token" viene fatto circolare automaticamente tra le stazioni master, ma non viene mai assegnato alle stazioni slave. Pertanto, queste ultime non possono leggere o scrivere dati da/su altre stazioni della rete.

### 11.5.5 Scambio dati tra stazioni Master e Slave



Uno scambio dati tra stazioni, controllato dall'utente, è ottenibile inserendo all'interno delle pagine Fupla gli appositi Fbox Fupla selezionabili mediante l'“Fbox Selector” (*Selettore Fbox*). Sono disponibili Fbox per scrivere (SEND) o leggere (RCV) pacchetti di dati, in grado di supportare anche vari formati di dati: binario, intero, virgola mobile, Data Block, ecc..

Gli Fbox *SEND* o *RCV* possono essere ridimensionati al fine di aumentarne o diminuirne il numero di ingressi ed uscite, definendo così le dimensioni del pacchetto di dati che può essere scambiato con un'altra stazione.

L'indirizzo del Canale di Comunicazione usato dall'Fbox di trasmissione dati è indicato dal simbolo posto nella parte superiore sinistra dell'Fbox che lo collega all'Fbox SASI avente nome identico e in cui è stato definito l'indirizzo del canale. Questo simbolo può essere modificato posizionando il cursore del mouse sull'Fbox e selezionando l'opzione “Fbox Properties – Name” (*Proprietà Fbox – Nome*) all'interno del menu contestuale.

Ciascun Fbox *SEND* e *RCV* è dotato di un ingresso binario per l'attivazione dello scambio dati. Se questo ingresso viene mantenuto costantemente al livello alto, lo scambio dati verrà ripetuto il più velocemente possibile. Inviando invece un breve impulso a questo ingresso, lo scambio dati verrà eseguito almeno una volta, ma è sempre possibile forzarne l'esecuzione usando il pulsante “Execute” (Esegui) oppure mediante una Ripartenza a Freddo del PCD, comandata con l'opzione “Initialization” (*Inizializzazione*) presente all'interno della finestra di configurazione (*adjust window*).

I dati della stazione Master presenti sugli ingressi dell'Fbox *SEND* sono inviati alla stazione Slave definita nella finestra di configurazione. Mentre i dati presenti sulle uscite dell'Fbox *RCV* giungono dalla stazione slave definita dai parametri specificati all'interno della finestra di configurazione: indirizzo della stazione slave, elemento sorgente e indirizzo di base.

Solo per le stazioni Master è possibile programmare gli Fbox *SEND* e *RCV*! Alle stazioni slave è consentito solo assegnare il canale di comunicazione.

In base agli Fbox usati, la *finestra di configurazione* permette di definire le stazioni slave alle quali la stazione master può inviare (*SEND*) i dati oppure le stazioni slave dalle quali la stazione Master può leggere (*RCV*) i dati.

#### **Parametri della Finestra di Configurazione**

##### ***IP Node (Nodo IP)***

Definisce il numero del nodo della stazione Slave Ether-S-Bus.

##### ***Source, destination station (Stazione Sorgente, Destinataria)***

Definisce il numero della stazione slave S-Bus.

##### ***Source, destination element (Elemento Sorgente, Destinatario)***

Definisce il tipo di dati da scrivere o leggere sulla/dalla stazione slave.

**Source, destination address (Indirizzo Sorgente, Destinatario)**

Definisce l'indirizzo di partenza del pacchetto di dati da scrivere o leggere sulla/dalla stazione slave. Il numero di valori di dati scambiati dipende dal numero di ingressi o di uscite dell'Fbox SEND o RCV.

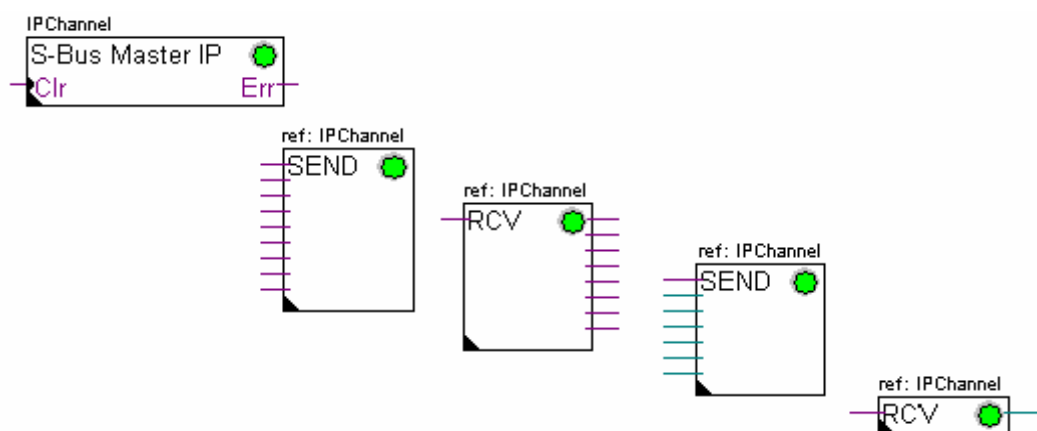
**11.5.6 Diagnostica**



Se il programma è "Online", nell'angolo superiore destro degli Fbox *SASI*, *SEND* o *RCV* viene visualizzato un LED di colore verde o rosso. Se questo LED è di colore verde, significa che la trasmissione dati viene eseguita correttamente (OK); se di colore rosso indica invece una condizione di errore.

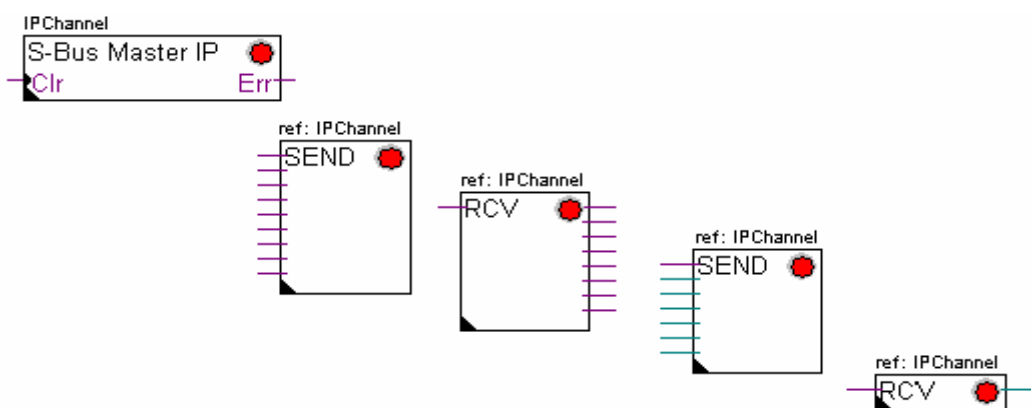
**Funzionamento corretto**

I LED di tutti gli Fbox sono di colore verde; lo scambio dati avviene correttamente.



**Nessun dato può essere scambiato all'interno della rete**

I LED dell'Fbox *SASI* e di tutti gli Fbox *SEND* e *RCV* sono di colore rosso; non è possibile trasferire alcun dato all'interno della rete.

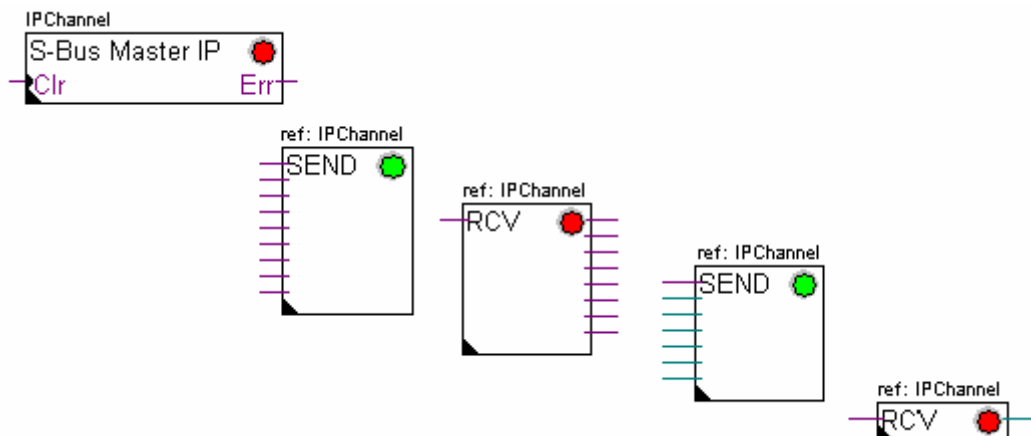


### Possibili azioni correttive eseguibili sulle stazioni master o slave:

- Verificare le impostazioni hardware all'interno della finestra "Hardware Settings"
- Verificare che le impostazioni hardware siano state correttamente caricate nel PCD
- Verificare che il canale di comunicazione definito con la finestra "Hardware Settings" sia identico a quello definito con la funzione SASI (stesso numero di canale)
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Ether-S-Bus

### Solo alcuni Fbox non effettuano lo scambio dati

I LED dell'Fbox SASI e di alcuni Fbox SEND e RCV sono di colore rosso; gli Fbox il cui LED è di colore verde eseguono correttamente lo scambio dati.



### Possibili azioni correttive eseguibili sulla stazione master

Verificare i parametri definiti all'interno della *finestra di configurazione* degli Fbox SEND e RCV il cui LED è di colore rosso.

Verificare che l'indirizzo della stazione slave specificato sia presente all'interno della rete.

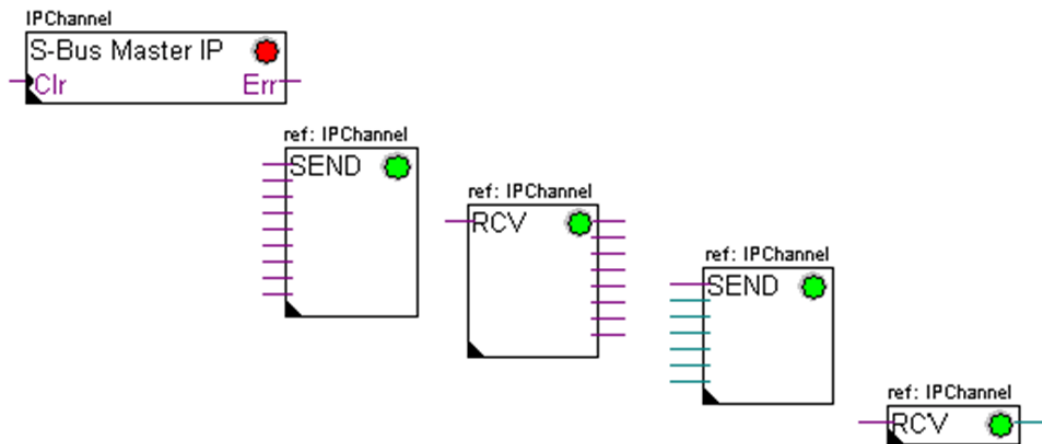
### Possibili azioni correttive eseguibili sulla stazione slave

Per tutti gli Fbox SEND e RCV il cui LED è di colore rosso, rilevare il numero di stazione slave specificato e verificare le corrispondenti stazioni.

- Verificare le impostazioni hardware all'interno della finestra "Hardware Settings"
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Ether-S-Bus

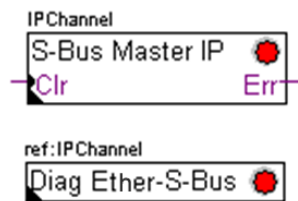
### **Solo il LED dell'Fbox SASI è di colore rosso**

Aprire la finestra di configurazione dell'Fbox *SASI* ed eliminare l'ultimo allarme per mezzo del pulsante "Clear".



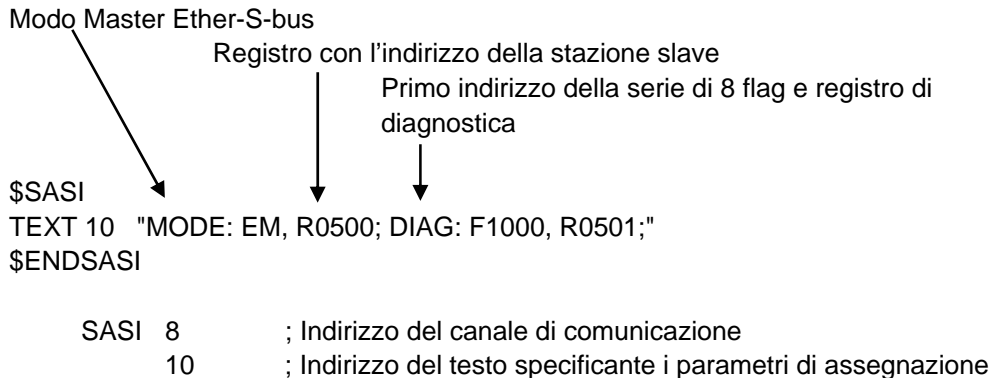
### **Fbox di Diagnostica**

Quando il LED dell'Fbox SASI è di colore rosso, è sempre possibile ottenere una diagnosi consultando la finestra di configurazione della funzione *Diagnostica SASI*. Il corrispondente Fbox deve essere inserito immediatamente sotto l'Fbox *SASI*.



## 11.6 Programma IL

### 11.6.1 Assegnazione del Canale Master usando l'istruzione SASI



L'assegnazione del canale viene effettuata usando l'istruzione SASI posta all'inizio del programma: nella sequenza di inizializzazione Graftec o nel blocco di inizializzazione XOB 16.

L'istruzione SASI contiene due parametri: l'indirizzo del canale di comunicazione e l'indirizzo del testo che definisce tutti i parametri del canale.

I parametri di assegnazione specificati via Testo sono diversi da una rete di comunicazione all'altra così come per le diverse stazioni slave o master.

Se il PCD usa più canali di comunicazione, è necessario definire ciascun canale usando un'istruzione SASI ed un Testo di assegnazione.

In base al tipo di rete adottato, la definizione dei parametri del canale può essere completata con la finestra *"Hardware Settings"* (*Impostazioni Hardware*).

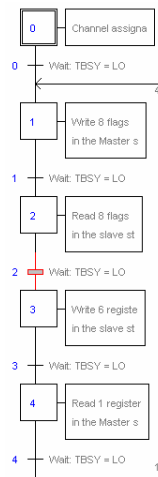
### 11.6.2 Assegnazione del Canale Slave

Per il Canale Slave di una rete Ether-S-Bus non è necessario definire alcuna istruzione SASI. Tutte le definizioni richieste sono già specificate all'interno della finestra *"Hardware Settings"*.

### 11.6.3 Principi legati allo scambio dati in una rete "multi-master"

Una rete di comunicazione "multi-master" integra più di una stazione "master". Le Stazioni Master sono le uniche autorizzate a leggere o scrivere i dati da/su le restanti stazioni master e slave. Lo scambio dati tra stazioni slave non è consentito. Con una modalità di comunicazione "Multi-master", lo scambio dati avviene sempre tra le stazioni master connesse alla rete. Solo ad una delle stazioni master per volta viene assegnato un "token" che la autorizza a scambiare i dati con le restanti stazioni master o slave in rete. Quando il master in oggetto ha ultimato il trasferimento dei dati, il "token" viene passato alla stazione master successiva che a questo punto è autorizzata a scambiare i dati con le altre stazioni master o slave. Il "token" viene fatto circolare automaticamente tra le stazioni master, ma non viene mai assegnato alle stazioni slave. Pertanto, queste ultime non possono leggere o scrivere dati da/su altre stazioni della rete.

### 11.6.4 Scambio dati tra stazioni Master e Slave



Passo Iniziale: assegnazione del canale

Passo: scambio dati

Transizione: attesa della fine dello scambio dati

Lo scambio dati tra le stazioni è un programma sequenziale: l'assegnazione del canale di comunicazione viene eseguita una sola volta e lo scambio dati all'interno della rete verrà effettuato solo se la precedente sessione di scambio dati è terminata. Per tali motivi si propone di definire la procedura di scambio dati in IL con l'Editore Graftec.

Il "Passo Iniziale" permette l'assegnazione del canale di comunicazione al Riavvio a Freddo del PCD.

I "Passi" successivi vengono eseguiti ciclicamente (loop) e un "passo" supporta lo scambio di un pacchetto di dati.

Ogni "passo" è separato da una "Transizione" che verifica lo stato del flag di diagnostica TBSY e stabilisce quindi se lo scambio dati è terminato. Si sarà autorizzati allo scambio dati definito dal "passo" seguente solo se il flag TBSY è allo stato "Basso".

#### **Scambio dati per mezzo di un Passo**

Prima di poter scambiare i dati, è necessario definire nel registro indirizzi dichiarato dal testo di assegnazione, l'indirizzo della stazione slave:

#### **Definizione dell'indirizzo della stazione slave**

LDL R 500 ; Registro indirizzi contenente l'indirizzo della stazione slave  
11 ; indirizzo S-Bus

LDH R 500 ; Registro indirizzi contenente l'indirizzo della stazione slave  
2 ; Nodo IP

Lo scambio dati tra le stazioni è ottenibile usando due istruzioni:

STXM: per la scrittura dei dati nella stazione slave (*SEND*)

SRXM: per la lettura dei dati dalla stazione slave (*RCV*)

Per ciascuna istruzione è necessario specificare quattro parametri: l'Indirizzo del Canale, il Numero di Dati da scambiare, l'Indirizzo di Base della Sorgente e Indirizzo di Base della Destinazione.



**Scrittura di 8 Flag (F 0... F 7) nella stazione slave (F 200... F 207)**

STXM 8 ; indirizzo del canale  
 8 ; numero di dati da scambiare  
 F 0 ; indirizzo di base della sorgente (stazione locale)  
 F 200 ; indirizzo di base della destinazione (stazione slave)

**Lettura di un registro (in R 125) dalla stazione slave (R 25)**

SRXM 8 ; indirizzo del canale  
 1 ; numero di dati da scambiare  
 R 25 ; indirizzo di base della sorgente (stazione slave)  
 R 125 ; indirizzo di base della destinazione (stazione locale)

Nota:

Solo le stazioni Master sono programmabili con le istruzioni STXM e SRXM ! Alle stazioni slave è consentito solo assegnare il canale di comunicazione.

**Attesa della “Fine Trasmissione” usando la Transizione**

STL F 1003 ; Verifica se TBSY si trova allo stato Basso

Il Testo di Assegnazione definisce una serie di 8 flag di diagnostica per la comunicazione. Il terzo di questi flag passa allo stato Alto durante la trasmissione dei dati ed allo stato Basso quando la sessione di scambio dati è terminata.

**11.6.5 Diagnostica****Assegnazione del canale**

In caso di problemi di comunicazione, verificare che l'assegnazione del canale venga eseguita correttamente. Analizzare il programma passo-passo e verificare che l'istruzione SASI non determini l'impostazione di un flag di errore. Se l'assegnazione del canale non viene eseguita in modo corretto, non sarà possibile stabilire la comunicazione.

**Possibili azioni correttive eseguibili sulle stazioni master o slave:**

- Verificare le impostazioni hardware all'interno della finestra “*Hardware Settings*”
- Verificare che le impostazioni hardware siano state correttamente caricate nel PCD
- Verificare che il canale di comunicazione definito con la finestra “*Hardware Settings*” sia identico a quello definito con l'istruzione SASI (stesso numero di canale)
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Ether-S-Bus

### **I dati non vengono scambiati all'interno della rete**

Il Testo di Assegnazione definisce una serie di 8 flag di diagnostica per la comunicazione. Il quinto di questi flag (*TDIA: Diagnostica Trasmettitore*) passa allo stato Alto in caso di errore durante la trasmissione dati. Testare passo-passo il programma per rilevare le istruzioni STXM e SRXM errate.

Attenzione: in caso di errore di comunicazione, il flag di diagnostica TDIA rimane nello stato Alto finché non viene reimpostato a zero (resettato) il registro di diagnostica.

#### **Possibili azioni correttive eseguibili sulla stazione master**

Verificare i parametri delle istruzioni STXM e SRXM che generano l'errore. Verificare che l'indirizzo della stazione slave specificato sia presente all'interno della rete.

#### **Possibili azioni correttive eseguibili sulla stazione slave**

Per tutte le istruzioni STXM e SRXM che generano un errore, rilevare il numero di stazione slave specificato e verificare le corrispondenti stazioni.

- Verificare le impostazioni hardware all'interno della finestra "*Hardware Settings*"
- Verificare che il PCD sia equipaggiato con l'hardware richiesto per la comunicazione
- Verificare che le stazioni siano collegate alla rete e che siano correttamente alimentate ed accese
- Verificare il cablaggio della rete
- Verificare che la versione di firmware usata supporti il protocollo Ether-S-Bus

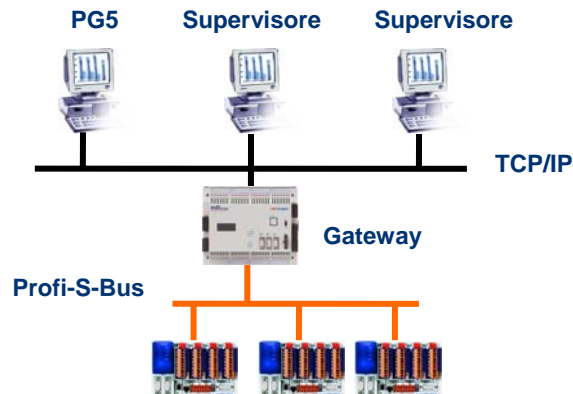
### **Registro di Diagnostica**

Il registro di diagnostica può fornire maggiori informazioni circa la natura dell'errore di comunicazione. Visualizzare il contenuto binario del registro e confrontarlo con le descrizioni riportate nel manuale del PCD o nel manuale dedicato alla Rete di Comunicazione utilizzata.

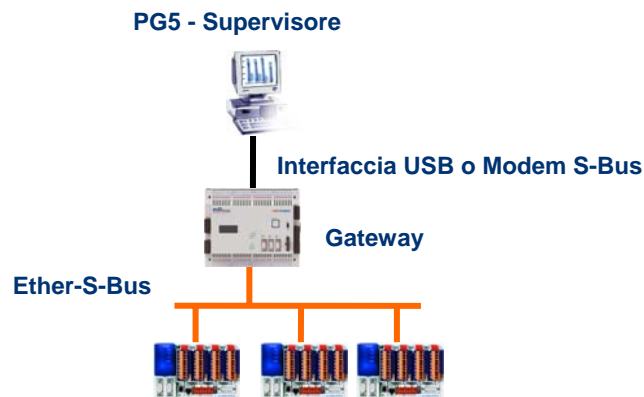
## 11.7 Funzione Gateway

La funzione *Gateway* viene tipicamente utilizzata per consentire a due reti di comunicazione diverse di comunicare tra loro oppure per usare il pacchetto di programmazione (PG5) o un sistema di supervisione (Visi+) con un tipo di rete diversa da quella normalmente supportata.

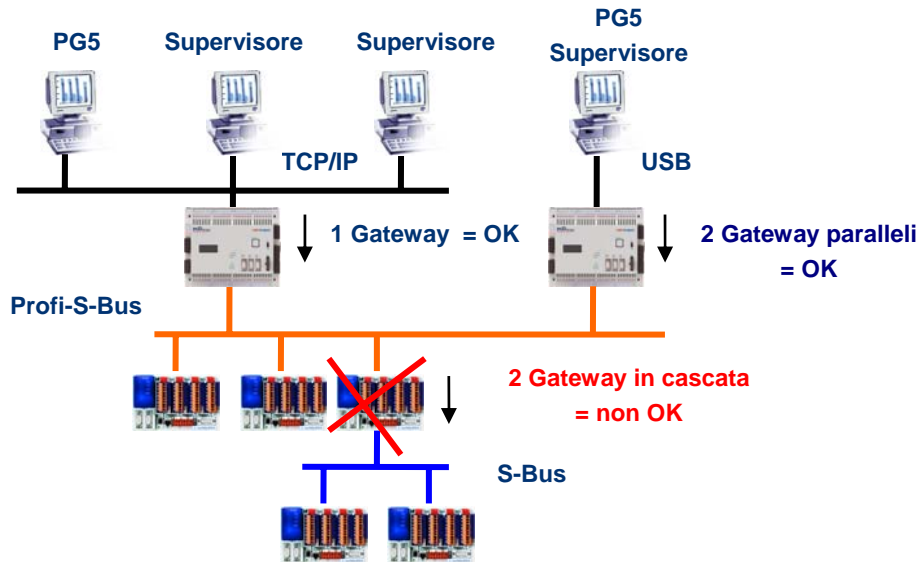
### 11.7.1 Applicazione



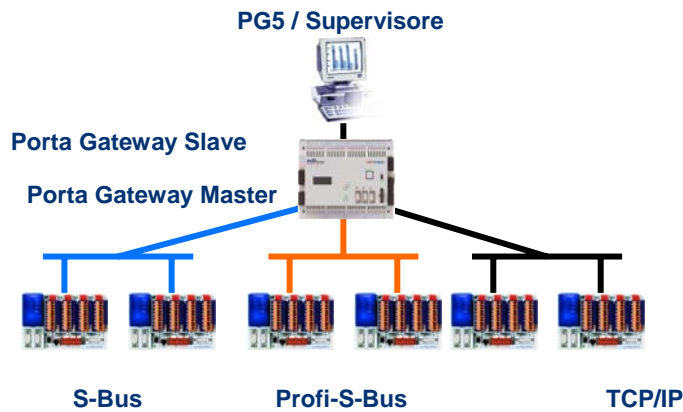
La funzione *Gateway* crea un "ponte" tra due reti consentendo, ad esempio, di collegare una rete Ethernet ad una rete Profi-S-Bus. In questo modo, i sistemi PCD possono scambiare dati attraverso un bus comune, specificatamente sviluppato per applicazioni di automazione, separato dalla rete informatica della società. Tuttavia, anche i PC su cui è installato il pacchetto PG5 o il sistema di supervisione Visi+ possono scambiare dati con i PCD.



La funzione *Gateway* può anche essere usata per creare un'interfaccia tra una rete di comunicazione ed il mondo esterno, realizzando, ad esempio, un'interfaccia di comunicazione USB o Modem.

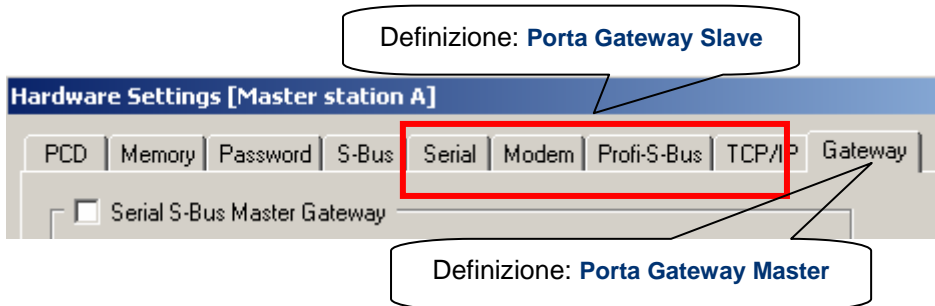


A causa della necessità di rispettare le temporizzazioni di comunicazione, non è possibile integrare due funzioni Gateway in cascata (serie). Tuttavia è possibile integrare due Gateway paralleli sulla stessa rete.



Quando necessario, un Gateway può essere usato per creare un “ponte” tra più sotto-reti di comunicazione.

### 11.7.2 Configurazione della funzione Gateway PGU

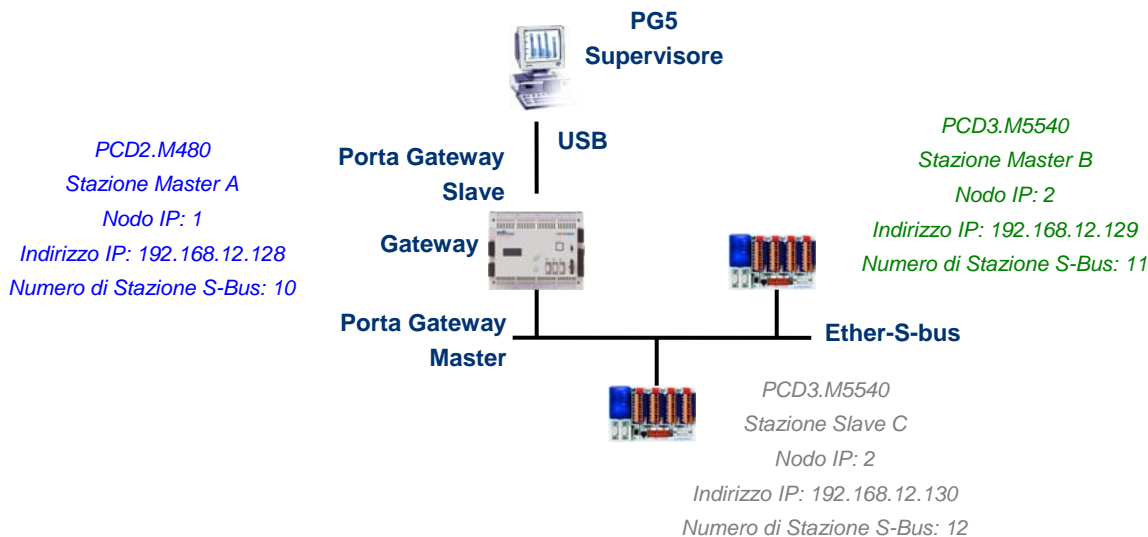


E' molto semplice configurare la funzione *Gateway*; non è infatti richiesta alcuna programmazione ma solo la definizione di alcuni parametri per il PCD all'interno della finestra “*Hardware Settings*” (*Impostazioni Hardware*).

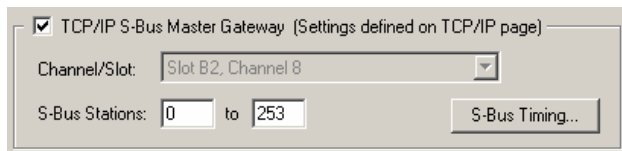
Generalmente, è necessario definire solo una Porta Gateway Slave e una Porta Gateway Master per abilitare automaticamente la funzione Gateway.

Se il messaggio ricevuto dalla *Porta Gateway Slave* non è destinato alla stazione locale (*il Gateway*), allora i dati vengono ri-trasmessi ad una delle sotto-reti collegate alla *Porta Gateway Master*, in base al campo di indirizzi definito per la sotto-rete in oggetto.

**Esempio: Gateway USB, Ether-S-Bus**

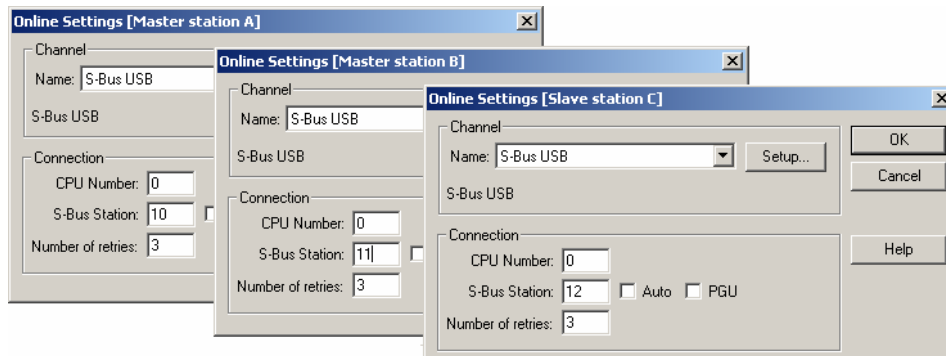


**Impostazioni Hardware per la Stazione Master A**




Il Gateway USB rappresenta un’eccezione, dato che non richiede la definizione di alcun parametro per la *Porta Gateway Slave* ma solo la definizione della *Porta Gateway Master* (non scordarsi di caricare la nuova configurazione sulla stazione Master A!).

**Impostazioni Online per le CPU usate nel progetto**



Per realizzare una comunicazione via USB con ciascun PCD, all’interno della finestra *“Online Settings” (Impostazioni Online)* è necessario selezionare il canale USB e specificare il numero di stazione S-Bus.


## Verifica della funzionalità Gateway

 Slave station C - PCD3.M5540 - IPNode 3, Station 12

Attivare una delle CPU *Master B* o *Slave C* del progetto e passare alla modalità "Online" per verificare la comunicazione con la stazione interessata.



Se necessario, l'"*Online Configurator*" (*Configuratore Online*) permette di verificare online il numero di stazione. E' anche possibile caricare il programma nella CPU attiva e, per verificarne il funzionamento, restare sempre collegati alla stazione *Master A* per mezzo di un cavo USB.

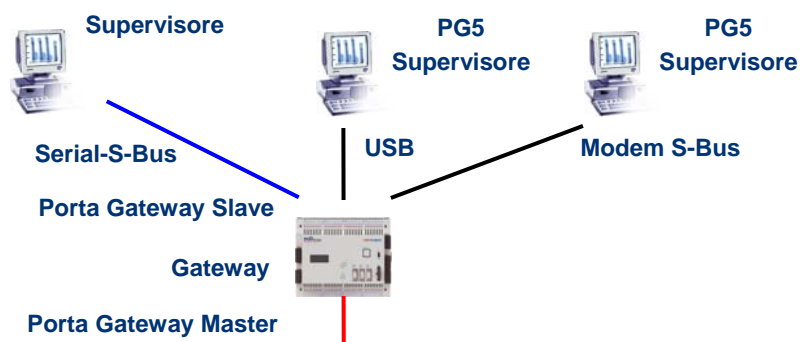
 Master station B - PCD3.M5540 - IPNode 2, Station 11

Per comunicare con un'altra stazione della rete, attivare la relativa CPU e passare alla modalità "Online".

Nota:

Con la funzione *Gateway*, viene considerato il solo numero di stazione S-Bus; il numero di stazione Ether-S-Bus viene ignorato dal momento che i telegrammi vengono indirizzati a tutte le stazioni Ether-S-Bus (distribuzione Broadcast).

### 11.7.3 Configurazione di una Porta Gateway Slave supplementare

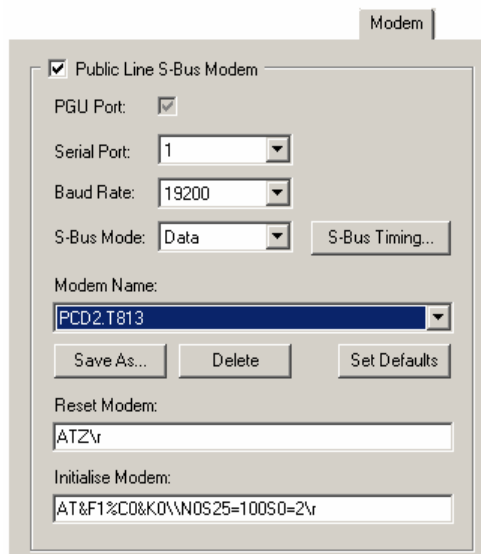


La Porta Gateway Slave rappresenta una via che permette di accedere alla rete dall'esterno. Se necessario, è possibile definire una seconda o una terza *Porta Gateway Slave*.

#### Impostazioni Hardware

Generalmente, i PCD supportano solo un canale PGU slave. Tuttavia, i nuovi controllori PCD2.M480 e PCD3.Mxxxx supportano l'uso di più di una porta PGU sullo stesso PCD. La configurazione della seconda PGU Gateway Slave è effettuabile all'interno della finestra "*Hardware Settings*" (*Impostazioni Hardware*).

**Esempio: aggiunta di un Gateway supplementare Modem S-Bus, Ether-S-Bus**



La seconda *Porta PGU Gateway Slave* è stata aggiunta configurando i parametri per il modem all'interno della finestra "Hardware Settings" (Impostazioni Hardware).

**Programmi Fupla o IL**

Con i PCD di vecchia generazione ma anche con i nuovi PCD2.M480 e PCD3.Mxxxx, è possibile usare un Fbox/Istruzione SASI supplementare per aggiungere una *Porta Gateway Slave* supplementare.

Questa *Porta Gateway Slave*, senza funzionalità PGU, non supporterà l'interazione con il pacchetto di programmazione PG5 ma solo con un terminale di supervisione. Inoltre, sarà ammessa solo la lettura/scrittura di dati PCD: registri, flag, ecc..

**Esempio Fupla: aggiunta di un Gateway supplementare Serial-S-Bus, Ether-S-Bus**

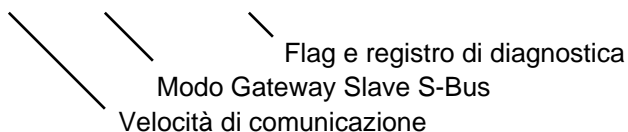


E' necessario impostare a "Yes" il parametro di configurazione "Gateway". In base al tipo di canale, devono poi essere correttamente definiti i vari parametri presenti nella finestra di configurazione dell'Fbox.

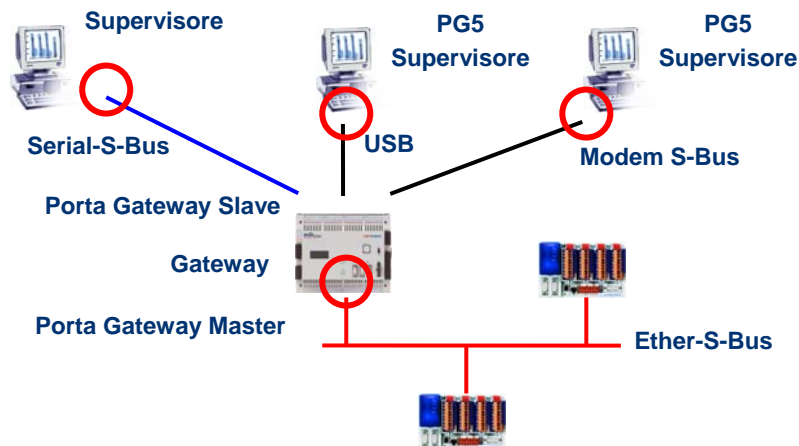
**Esempio IL: aggiunta di un Gateway supplementare Serial-S-Bus, Ether-S-Bus**

Usare il seguente Testo per assegnare il canale:

```
$SASI
TEXT 11 "UART:9600; MODE:GS2; DIAG:F1110, R0501;"
$ENDSASI
```

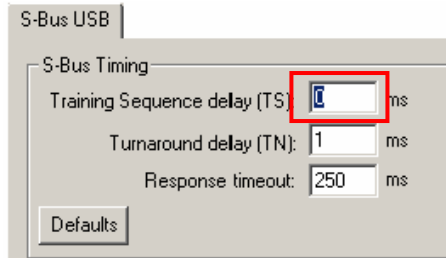


### 11.7.4 Temporizzazioni di Comunicazione



Generalmente, le *temporizzazioni* di comunicazione sono impostate ai valori di default e tale impostazione permette un funzionamento corretto del sistema. Tuttavia, implementando la funzione *Gateway* si aumentano i tempi di reazione richiesti per lo scambio dati. Per tale motivo, a volte è necessario modificare l'intervallo di timeout delle stazioni master che usano la funzione *Gateway*. L'immagine precedente illustra quali sono i canali Master che richiedono una personalizzazione del timeout.

Per regolare il *Timeout* PG5, usare la finestra "*Online Settings*" (*Impostazioni Online*) associata alla *Stazione Master A*:



Per regolare il *Timeout* del programma di scambio dati del PCD, usare l'Fbox *SAS/ S-Bus IP Extended*



### 11.8 Ulteriori Informazioni/riferimenti

Per maggiori dettagli, è possibile consultare i seguenti manuali/riferimenti:

- Manuale delle Istruzioni 26/133
- Manuale Ethernet TCP/IP 27/776
- Esempio di progetto Ether-S-Bus installato con il pacchetto PG5



## Indice

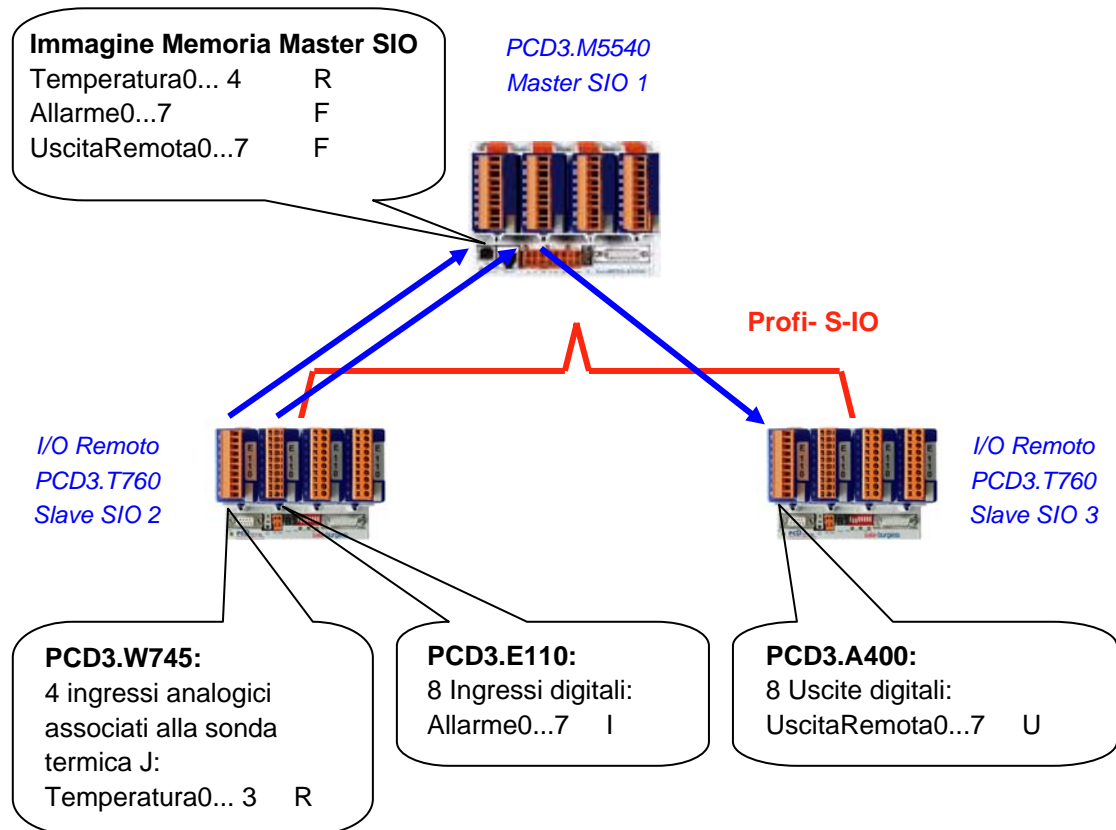
---

<b>12</b>	<b>Profi-S-IO.....</b>	<b>2</b>
12.1	<i>Esempio di rete Profi-S-IO .....</i>	2
12.2	<i>Generalità sul funzionamento.....</i>	2
12.3	<i>Il Progetto PG5 .....</i>	3
12.4	<i>Definizione delle stazioni connesse alla rete.....</i>	3
12.5	<i>Configurazione della Stazione Master.....</i>	4
12.6	<i>Configurazione delle Stazioni Slave .....</i>	4
12.6.1	<i>Configurazione dei moduli di Ingresso/Uscita .....</i>	4
12.6.2	<i>Configurazione del nome dei Simboli associati ai Dati Remoti .....</i>	5
12.6.3	<i>Configurazione dei parametri di I/O.....</i>	5
12.7	<i>Configurazione della rete .....</i>	6
12.8	<i>Uso dei Simboli di Rete nei programmi Fupla o IL.....</i>	6
12.9	<i>Ulteriori Informazioni/riferimenti .....</i>	7

## 12 Profi-S-IO

Questo esempio illustra come utilizzare gli ingressi e le uscite remote sia digitali che analogiche di un PCD3.T7xx RIO.

### 12.1 Esempio di rete Profi-S-IO



### 12.2 Generalità sul funzionamento

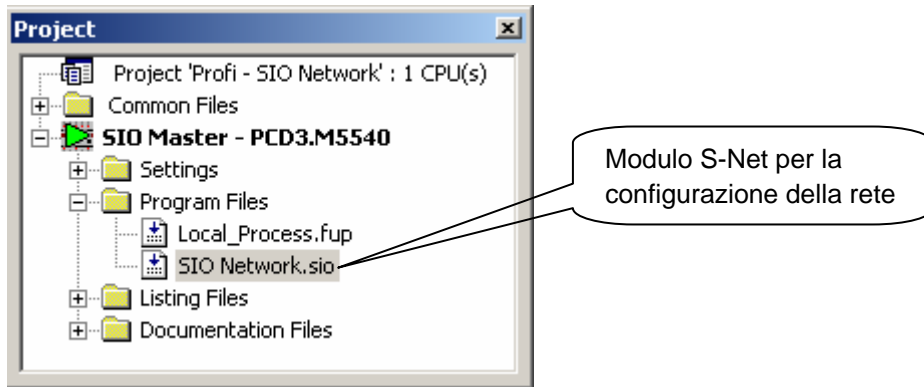
Con entrambi i protocolli Profibus DP e Profibus-S-IO, lo scambio dati all'interno della rete può essere definito con il Configuratore S-Net. Non è necessario scrivere alcun codice Fupla o IL né definire alcuna speciale impostazione hardware mediante l'apposita finestra "Hardware Settings" (eccezion fatta per il tipo di modulo di comunicazione e i parametri del bus in caso di utilizzo del PCD2.M480 o del PCD3.Mxxxx).

Il configuratore permette di definire ciascuna stazione slave connessa alla rete e quali moduli di I/O sono presenti. I dati di questi dispositivi di I/O remoto vengono riferiti (mappati) a simboli o indirizzi assoluti all'interno della stazione master. Il codice generato dal configuratore S-Net determina il trasferimento continuo dei dati di I/O dalle stazioni slave da/verso l'immagine di memoria del master.

Alla compilazione del programma, S-Net genera tutto il codice necessario per ottenere, all'avvio/fine di ogni ciclo, il trasferimento continuo dei dati tra le stazioni slave remote e l'immagine di memoria della stazione master. I programmi Fupla o IL della stazione master possono accedere direttamente ai dati di immagine I/O.

In questo modo, lo scambio dei dati all'interno della rete risulterà chiaramente separato dal controllo di processo.

### 12.3 Il Progetto PG5

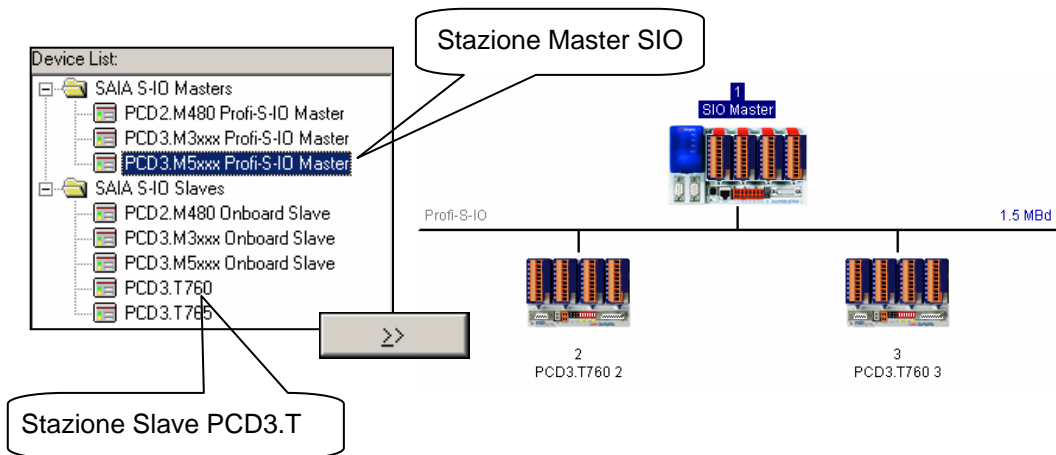


Il file del Configuratore S-Net può essere aggiunto alla stazione master come se si trattasse di un qualsiasi file Fupla o IL, usando il comando *File – New (File – Nuovo)* e selezionando il tipo di file "Profi-S-IO Network File (.sio)".

L'uso del Configuratore S-Net è simile per entrambe le funzioni di scambio dati basate su Profi-S-IO e Profibus DP. Le uniche differenze sono:

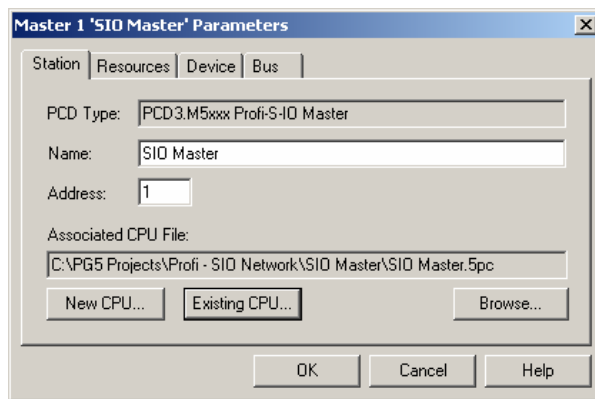
- L'estensione del file di configurazione: .SIO, .DP
- I dispositivi supportati nella rete: SIO = dispositivi Saia, DP = dispositivi per Saia + altri costruttori.
- Profili di temporizzazioni bus: S-Net o DP.

### 12.4 Definizione delle stazioni connesse alla rete



Per ogni singola stazione connessa alla rete, selezionarne il relativo tipo all'interno dell'elenco dispositivi (Device List) quindi aggiungerla alla rete con il pulsante >>.

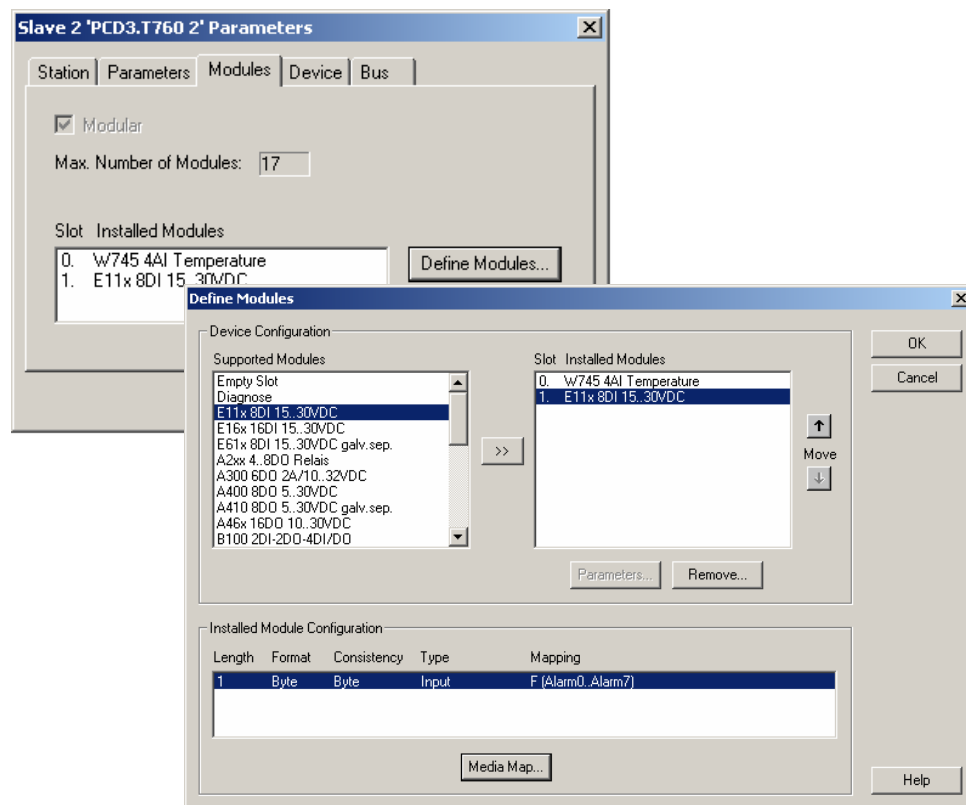
## 12.5 Configurazione della Stazione Master



L'unica informazione richiesta per definire la Stazione Master è il parametro "Associated CPU File" (*File CPU Associato*), che in pratica è il percorso di accesso al file della CPU Master. Questa è la posizione in cui il configuratore S-Net creerà il file di controllo della rete per la stazione master. La finestra di dialogo in oggetto permette anche di definire il nome della stazione (campo "Name") ed il relativo indirizzo (campo "Address").

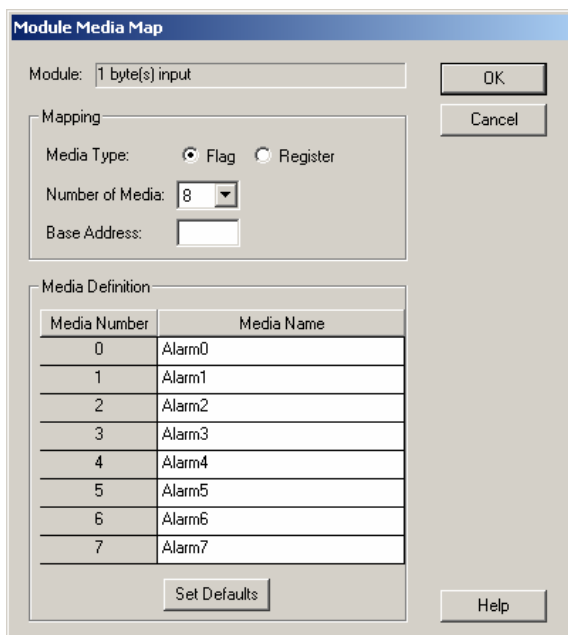
## 12.6 Configurazione delle Stazioni Slave

### 12.6.1 Configurazione dei moduli di Ingresso/Uscita



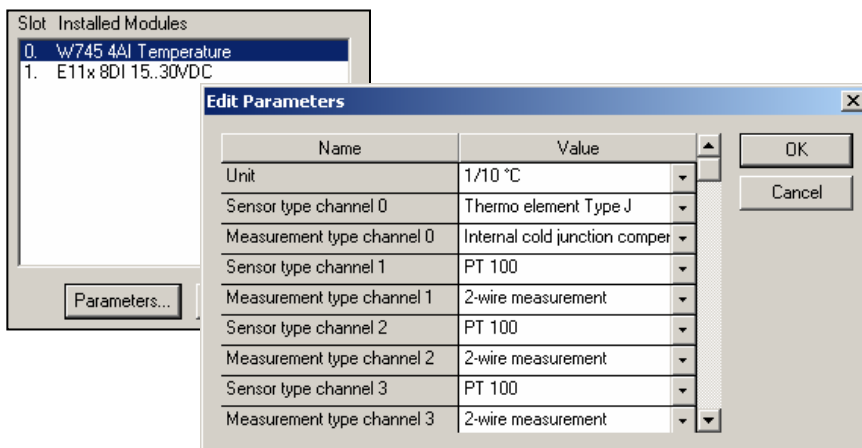
Per ciascun modulo di ingresso/uscita installato sulla stazione slave, selezionarne il relativo tipo all'interno dell'elenco "Supported Modules" (*Moduli Supportati*) ed aggiungerlo all'elenco "Installed Modules" (*Moduli Installati*) usando il pulsante >>. Accertarsi che il numero di "Slot" indicato corrisponda allo slot in cui è effettivamente installato il modulo in oggetto. Se necessario, riordinare l'elenco (modificare il numero di slot associato) usando i tasti freccia Su/Giù.

### 12.6.2 Configurazione del nome dei Simboli associati ai Dati Remoti



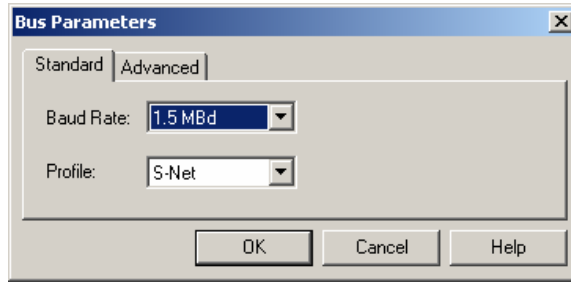
Selezionare ogni singolo modulo presente all'interno dell'elenco "Installed Modules" quindi fare clic sul pulsante "Media Map..." (Mappa Elementi) per definire i nomi dei Simboli ed i tipi di Elementi da associare ai dati del modulo stesso. Se necessario, è possibile definire un indirizzo di base (Base Address) per il primo flag o registro all'interno della stazione master. Tuttavia, la soluzione più semplice è quella di lasciare il campo "Base Address" vuoto in modo che il sistema utilizzi gli indirizzi dinamici.

### 12.6.3 Configurazione dei parametri di I/O



Per alcuni moduli, quali ad esempio i moduli di misura analogici, è necessario definire alcuni parametri supplementari per procedere alla selezione delle unità di misura, del tipo di sensore, ecc.. Questi parametri supplementari sono configurabili selezionando il modulo interessato e facendo clic sul pulsante "Parameters..." (Parametri).

## 12.7 Configurazione della rete

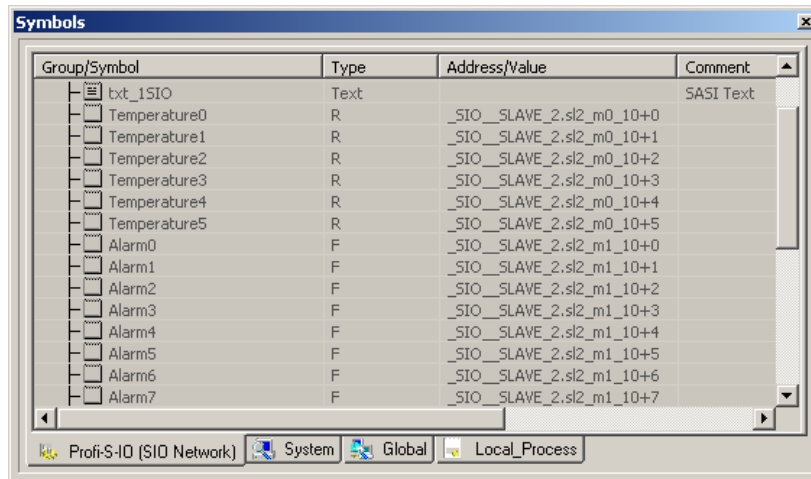


La velocità di comunicazione ed i profili dei bus sono selezionabili usando la voce di menu *“Edit - Bus Parameters”* (*Modifica – Parametri Bus*).

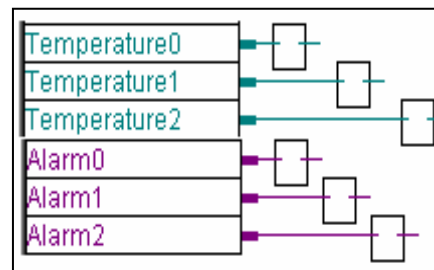
Nota:

Se alla rete è collegata una stazione PCD7.T7xx, selezionare sempre il profilo bus *“S-Net”*.

## 12.8 Uso dei Simboli di Rete nei programmi Fupla o IL



<b>LD</b>	<b>Temperature0</b>
<b>STH</b>	<b>Alarm0</b>
<b>OUT</b>	<b>RemoteOutput1</b>



Dopo la compilazione del file S-Net (ottenibile usando la voce di menu *“Project / Compile”* (*Progetto / Compila*)), all’interno del *“Symbol Editor”* (Editore dei Simboli) viene visualizzata una nuova pagina contenente tutti i simboli di rete accessibili. Questi simboli possono essere usati direttamente all’interno di programmi Fupla ed IL.

## 12.9 Ulteriori Informazioni/riferimenti

Per maggiori dettagli, è possibile consultare i seguenti manuali/riferimenti:

- Manuale Profibus DP 26/ 765
- Manuale Profi-S-IO (in preparazione)
- Esempio di progetto Profi-S-IO installato con il pacchetto PG5

## Dati tecnici e specifiche per le ordinazioni

### Dati tecnici

Sistema operativo	Windows 95 B Windows 98 seconda edizione Windows NT 4.0 SP5 Windows 2000 deve essere installata l'interfaccia TCP/IP devono essere installate le librerie TAPI 2.0 o superiori
PC IBM compatibile	Pentium 150 o superiore; minimo 32 MB di RAM; 30 MB di spazio libero su hard disk; lettore di CD-ROM
Repertorio istruzioni	PCD Sono supportate tutte le 150 istruzioni PCD
FBox standard	Il pacchetto PG5 possiede oltre 250 FBox standard
Modem	Nel pacchetto PG5 sono implementate una configurazione e una comunicazione modem di base. Sono disponibili anche librerie con funzioni modem estese, quali SMS e Pagers
Linguaggi di programmazione	Lista Istruzioni (IL), FUPLA (FBD) e GRAFTEC (SFC)
CPU supportate	Sono supportati tutti i modelli SAIA® PCD (ad esclusione della serie xx7)
Compatibilità	I programmi PG3 e PG4 si possono ancora utilizzare con PG5
Comunicazione	In PG5 sono disponibili le comunicazioni TCP/IP, SAIA® S-Bus, PROFIBUS DP, PROFIBUS FMS e LONWORKS®.

### Saia-Burgess Milano Srl

Via Cadamosto 3  
I-20094 Corsico, Milano  
Telefono (02) 48 69 21  
Facsimile (02) 48 60 06 92  
E-mail: [saia-burgess.it@saia-burgess.com](mailto:saia-burgess.it@saia-burgess.com)  
Homepage: [www.saia-burgess.com](http://www.saia-burgess.com)  
Supporto: [www.sbc-support.ch](http://www.sbc-support.ch)

### Specifiche per le ordinazioni

Tipo	Descrizione
<b>PCD8.P59 000 M9</b>	<b>Pacchetto software PG5 completo</b> Il pacchetto contiene un dischetto con la licenza d'uso, la documentazione ed il programma su CD-ROM.
<b>PCD8.P59 000 M1</b>	<b>Pacchetto software demo PG5</b> Il pacchetto contiene la versione completa di PG5, ma è stata disabilitata la stampa dei file di programma e l'elaborazione è stata limitata a programmi di dimensioni inferiori a 2000 righe.