

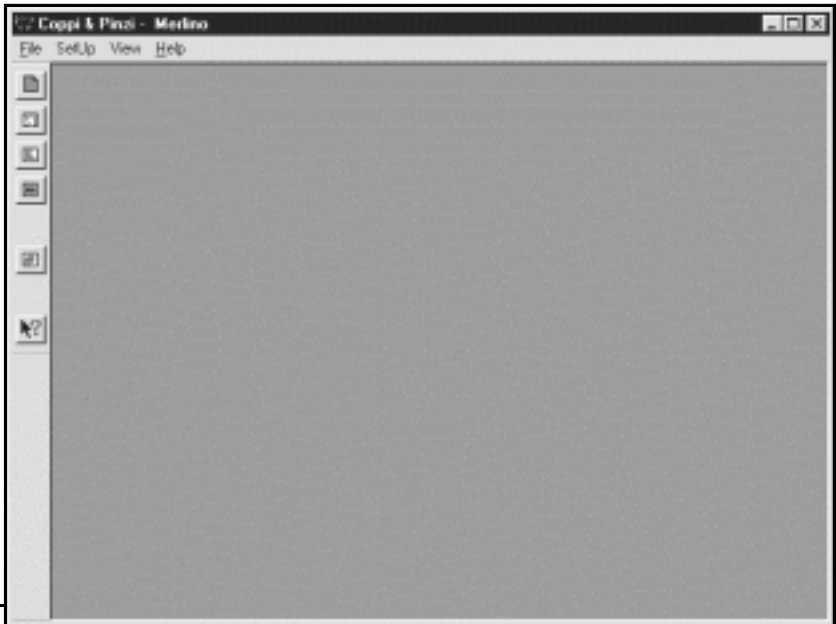
MANUALE
UTENTE
SOFTWARE
MERLINO.EXE
PER
WINDOWS 95
VERSIONE 2.00

AGGIORNAMENTI ALLE NUOVE VERSIONI

OTTOBRE 1999 – Vers. 2.76

Questo libro riguarda la progettazione software sul programma Merlino.exe V2.00 per il MicroPLC Merlino della C&P. Il volume non pretende di rappresentare tutte le possibilità di programmazione eseguibili ma di illustrare in maniera chiara i circuiti fondamentali e la tecnica di programmazione. Il testo viene sviluppato per esempi in maniera didattica in modo da istruire in maniera semplice. Si suppone che il programma sia stato installato in maniera corretta e sia pronto ad eseguire i comandi.

Quando si apre il programma merlino.exe si ha la seguente figura:



L'ambiente è pronto per editare un programma su schema elettrico ,con lista di istruzioni oppure a caricare un programma eseguibile sul PLC .

Per addentrarci nel sistema inizieremo a eseguire programmi da schema elettrico.

Per iniziare impostiamo le righe di visualizzazione della pagina dove viene disegnato lo schema elettrico.Selezionare nel menu la voce **SetUp e Impostazioni**.Compare il seguente dialog Box:



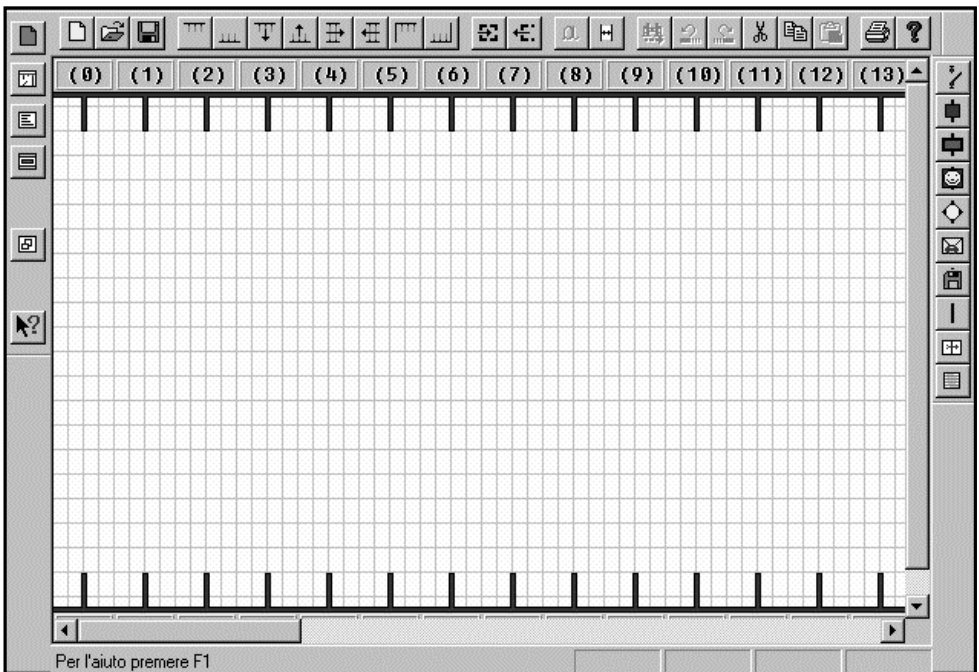
Nell'edit **Y ampiezza foglio disegno** impostare se si lavora a 640x480 pixel 8 mentre 10 se si ha risoluzione 800x600.Negli altri campi si può impostare i propri dati anagrafici di modo che questi siano inclusi quando si esegue una stampa di uno schema.Dare OK.

Possiamo Aprire un foglio per la programmazione del PLC tramite disegno Elettrico.

Nella barra principale posta a sinistra selezionare il bottone Nuovo Disegno:



Compare il foglio di lavoro con i propri toolbar.



Il toolbar superiore implementa i seguenti comandi:



Nuovo Disegno



Aprire un disegno



Salva il disegno corrente



Spostamenti sul foglio di lavoro



Macro importa esporta



Controllo Ortografico



Nuova colonna o riga



Stampa



Find,Undo,Redo,Cut,Copy,Paste

Il toolbar di sinistra serve a selezionare gli strumenti da disegnare .Ai primi 6 bottoni è associato un toolbar dedicato ai componenti di questa specie:





Dando un sommario sguardo ai vari componenti vediamo che si possono distinguere essenzialmente in due specie,contatti e bobine.

Con il termine **Contatti** intendiamo un contatto che può essere associato ad un ingresso ad un componente interno (rele ,timer etc) o ad un test è comunque sempre un dispositivo di ingresso.

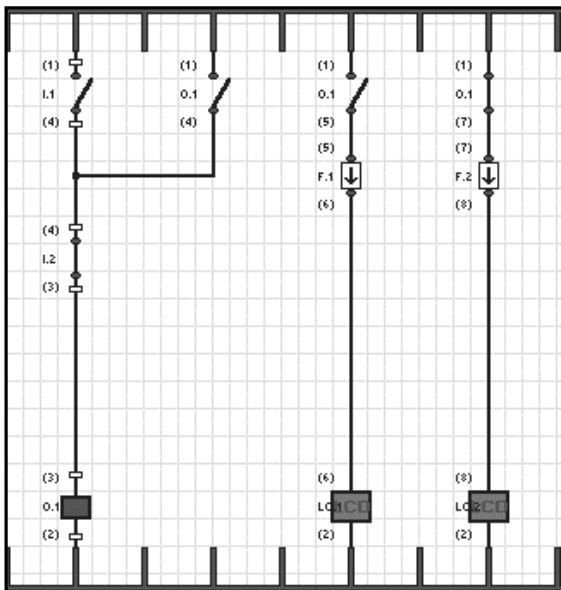
Con il termine **Bobina** intendiamo un dispositivo di uscita come ad esempio per la scrittura su display, o la scrittura sulla porta seriale nonché l'accensione vera e propria di un relé.

La programmazione con schema elettrico è la implementazione di uno schema funzionale il cui funzionamento riproduce gli stati richiesti.Sarà il compilatore e il PLC a simulare via software i blocchi funzionali virtuali in modo da dare delle risposte ai vari stati in modo del tutto analogo allo schema elettrico realizzato .Risulta quindi essenziale che per programmare con questo sistema bisogna avere una certa familiarità con i schemi elettrici di comando.

Passiamo quindi ad implementare il nostro primo programma:

Vogliamo realizzare un circuito START e STOP e quindi un circuito ad autoritenuta con UN ingresso di start, un ingresso di stop una uscita che rispecchia lo stato attuale ed inoltre due pagine associate che indicano sul display lo stato dell'uscita.

Realizziamo quindi il seguente schema elettrico:



Possiamo così descrivere il circuito:

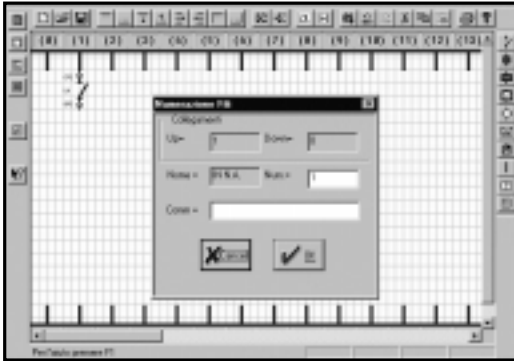
Se si preme I.1 O.1 si attiva perché I.2 è Normalmente Chiuso.Ma in parallelo a I.1 vi è un contatto Normalmente aperto di O.1 che si chiude per cui anche rilasciando I.1 per il parallelo dei contatti O.1 rimane attivo(circuito di memoria).L'unico modo per disattivare O.1 è premere I.2 in questo modo il contatto normalmente chiuso di I.2 si apre e O.1 si disattiva.Per visualizzare i messaggi di start e stop sono usate due bobine display con i messaggi impostati in START e STOP.

Quando O.1 è spento il contatto normalmente chiuso di O1 con in serie F2 è attivo per una scansione di programma e imposta il messaggio STOP.Quando si attiva O.1 in contatto normalmente aperto con in serie F.1 si attiva per una scansione di programma impostando il messaggio START.

COME REALIZZARE QUESTO ESEMPIO

Per prima cosa preleveremo un contatto di ingresso normalmente aperto. Selezionando il primo bottone sul toolbar di sinistra compare il toolbar di selezione componenti inpostato sui contatti.I contatti di ingresso si

differenziano dai contatti generici perché hanno dei piccoli rettangoli blu ai due estremi superiore ed inferiore. Per prelevare il contatto (ed ogni tipo di strumento) è sufficiente posizionarsi al centro con il mouse e poi cliccare una volta. Dopo questa operazione la barra degli strumenti diventerà completamente grigia. A questo punto spostare la freccia sotto alla colonna 1 e cliccare con il pulsante sinistro del mouse. In questo modo avremo piazzato il primo componente (il contatto di ingresso NA). Cliccando due volte con il pulsante sinistro sul componente installato si potrà accedere al dialog di impostazione dati come raffigurato nella figura sottostante.

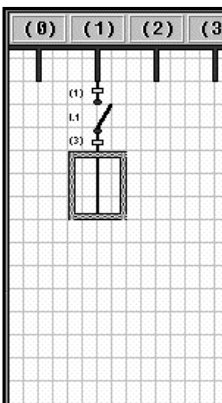


All'interno di questo dialog è possibile inserire il numero di ingresso associato al contatto ed un commento che identificherà la funzione del contatto stesso (per esempio START).

Adesso bisognerà inserire un filo verticale sotto all'ingresso N°1. Il filo può essere prelevato dalla solita barra strumenti e con le stesse modalità spiegate precedentemente. Adesso portarsi con il puntatore del mouse sotto al contatto precedentemente inserito e circa a metà del rettangolo evidenziato.

A questo punto cliccando con il pulsante sinistro del mouse otterrete il piazzamento

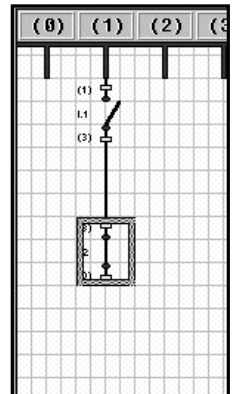
del filo verticale come raffigurato nella seguente figura:

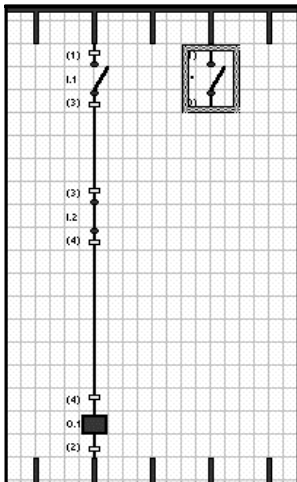


Adesso è possibile prelevare il contatto normalmente chiuso e posizionarlo sotto al filo verticale procedendo sempre allo stesso modo. Il risultato ottenuto sarà quello rappresentato nel disegno raffigurato sotto a destra.

Dopo aver inserito il numero di ingresso ed il relativo commento come per l'ingresso 1 possiamo passare all'inserimento della bobina di uscita. Come per gli ingressi, le uscite si differenziano dalle bobine generiche perché contraddistinte alle estremità da due rettangolini blu. Prelevare per mezzo del mouse la bobina uscita, portarsi immediatamente sotto all'ingresso 2 (nella casella vuota raffigurata), premere il pulsante sinistro del mouse.

Il risultato sarà il piazzamento della bobina in fondo al disegno con l'aggiunta automatica di un nuovo filo. L'ultimo componente da piazzare è il contatto di automantenimento **01** in parallelo all'ingresso N°1. Questo contatto può essere prelevato cliccando sul contatto NA interno generico (il primo a sinistra a sinistra nel tool selezione componenti contatti), e, utilizzando le stesse modalità precedenti può essere collocato nella colonna N° 3 in alto a destra di **II**





come rappresentato nel disegno qui a fianco. Il contatto così piazzato non corrisponde a nessun componente se non si provvede ad assegnargli un nome ed un indirizzo, per far questo è necessario, come per gli ingressi, cliccare due volte sul componente con il pulsante sinistro del mouse sino a veder comparire sul video la finestra di dialogo qui sotto raffigurata. Con questo dialog è possibile impostare il tipo ed il numero di contatto interno che interessa.



Per selezionare il contatto della **uscita N°1** è sufficiente cliccare sul cerchietto **Out** ed inserire il N° **1** nella casella **NUM=**.

Convalidando con il pulsante **OK** otterremo l'assegnazione al contatto interno precedentemente inserito.

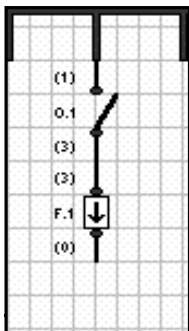
Adesso il contatto sullo schema elettrico prende il nome **O.1**

Adesso bisogna collegare il contatto all'ingresso N°1 così da poter espletare alla funzione di autoritenuta dello schema generale.

Per fare questa operazione bisognerà tenere conto di una regola molto importante che varrà poi anche per i collegamenti verticali più lunghi di un singolo filo.

IMPORTANTE : Per il prolungamento di qualunque tipo di filo sia in senso orizzontale che verticale non è necessario utilizzare altri fili ma solo allungare quello esistente procedendo nella maniera seguente:

- **Prolungamento di un filo verticale:** *cliccare sul filo interessato, la freccia del puntatore mouse si traformerà in una matita, portarsi ora con il mouse in corrispondenza del componente da collegare (meglio se leggermente all'interno del componente stesso) e cliccare nuovamente, si vedrà il filo allungarsi dal componente superiore, dove era stato collocato, al componente da collegare.*
- **Prolungamento di un filo verticale con diramazione verso destra:** *cliccare sul filo interessato, la freccia del puntatore mouse si traformerà in una matita, portarsi ora con il mouse in orizzontale verso destra e sotto al componente da collegare, cliccare nuovamente per vedere il filo che si allunga sino alla matita. Eseguire nuovamente la procedura dall'ultima posizione raggiunta e spostarsi verso l'alto sotto al componente interessato (meglio se leggermente all'interno del componente stesso) per cliccare nuovamente e vedere finalmente l'ultima parte del filo collegarsi al componente.*



Una volta ricordate queste regole il disegno risulterà semplice e rapido da realizzare. Adesso dobbiamo eseguire i due rami per la visualizzazione delle pagine. Prendere dai componenti un contatto generico normalmente aperto e impostarlo come visto in precedenza su uscita N°1 **O.1**. Dal barra strumenti per i contatti prendere un rilevatore di fronte:



Un rilevatore di fronte è un congegno che trasforma uno stato stabile in un impulso, quindi se il contatto normalmente aperto di I.1 si chiude il fronte F.1 farà passare solo un impulso fino a che I.1 non si apre.

Piazzare il rilevatore di fronte subito sotto il contatto N.A. di O.1 realizzando la seguente parte di schema:

Adesso bisogna piazzare la bobina di visualizzazione :

Dalla barra strumenti display selezionare una bobina LCD:

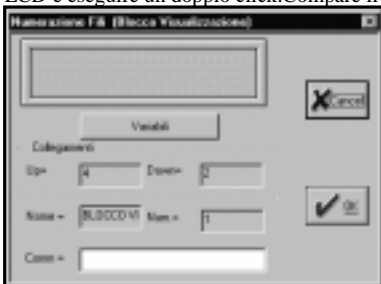


Una bobina LCD è una bobina che quando è abilitata costruisce e visualizza una pagina che comprende testo e variabili. Ogni volta che è abilitata scrive sia il testo che lo stato delle variabili.

Adesso risulterà chiaro perché si è dovuto mettere un fronte, in caso contrario la pagina verrebbe riscritta in continuazione visualizzando un leggero tremolio fastidioso alla lettura.

La bobina LCD è forse uno dei componenti più complessi che mette a disposizione il compilatore per schemi elettrici per cui in questa prima sede daremo solo un breve sguardo al dialog di impostazione di questa bobina rimandando più avanti nel testo ad un approfondimento di tale argomento.

Portarsi con il mouse subito sotto al rilevatore di fronte e piazzare la bobina LCD. Portarsi sopra alla bobina LCD e eseguire un doppio click. Compare il seguente dialog di impostazione:



Portarsi sopra la zona Verde del display con il mouse e eseguire un click con il tasto sinistro del mouse. Compare un cursore nero basso sul display virtuale. Con la tastiera digitare la parola "START". Sul display virtuale compare la scritta START.

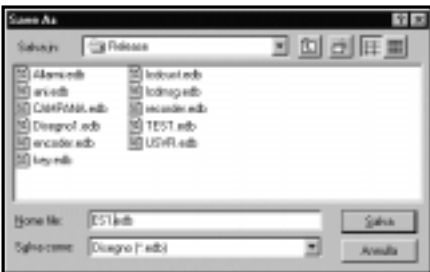


Adesso la bobina è già impostata sul messaggio richiesto. Dare OK.

Adesso la prima colonna di visualizzazione è terminata. Passiamo alla costruzione della seconda. Analogamente prendiamo un contatto generico, ma questa volta normalmente chiuso, e impostiamolo sulla uscita N° 1 O.1, sotto immettiamo il fronte F.2 e infine piazziamo la bobina LCD con il messaggio STOP.

Con questa ultima procedura abbiamo realizzato il nostro primo disegno che ora è pronto per essere compilato e inviato al PLC.

Prima di compilare questo disegno bisognerà salvarlo selezionando *File, Salva con nome*, dal menu in alto. Sarà poi sufficiente inserire il nome del file ricordandosi di mantenere l'estensione .EDB, che sta ad indicare che quello salvato è un file disegno, e facendo attenzione a collocarlo all'interno della sottodirectory desiderata. Tutte queste procedure seguono le stesse modalità di tutti i programmi che funzionano in ambiente WIN95.



Per passare alla fase di compilazione è sufficiente cliccare sull'ultimo strumento in basso a destra raffigurante un foglio:



Compila il disegno Corrente

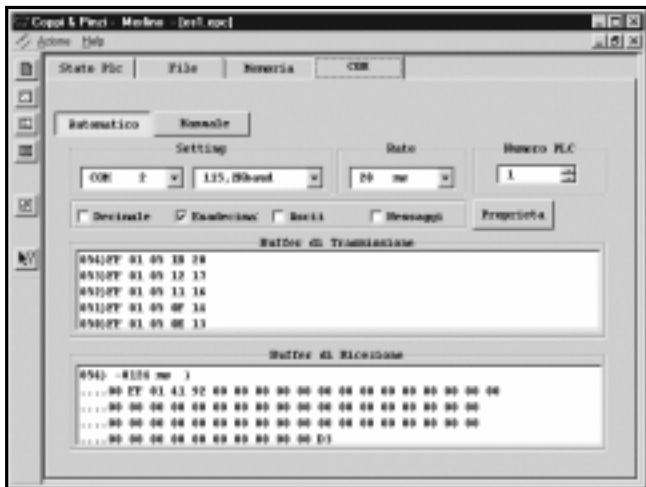
Inviando questo comando compare il seguente dialog:

Qui è possibile inserire il nome del file compilato (file avente estensione .EPC, il nome del file List (contenente il linguaggio macchina) ed il nome del file assembler avente estensione .ESM).



Il campo *Esegui Programma di Simulazione dopo la compilazione* non è abilitato di default

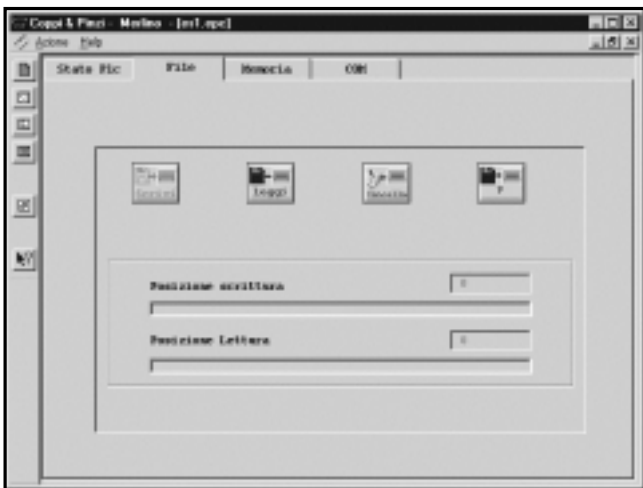
selezionare questo campo e dare conferma sul tasto OK. Viene eseguita la fase di compilazione. Se è stato effettuata tutto correttamente sarà minimizzata la pagina disegno, generata una finestra contenente il sorgente assembler del disegno e aperta una finestra di simulazione con il codice eseguibile del programma. Quella riportata sotto è la finestra che compare a compilazione avvenuta.



Se si è eseguito in modo corretto la installazione del PLC come riportato nel manuale utente nella finestra di figura sarà un susseguirsi di numeri in rapido scorrimento. Questo sta a significare che la comunicazione tra personal computer e PLC avviene in modo appropriato.

Selezionare la cartella **File**

Compare il nuovo dialog di figura:



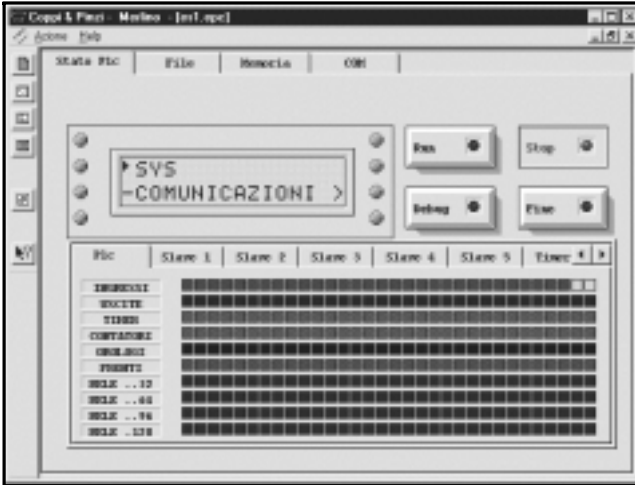
Se il PLC non ha nessun programma caricato il tasto Scrivi è abilitato in caso contrario è abilitato il tasto cancella. In questo ultimo caso inviare un comando cancella. Il PLC libera la memoria occupata e dopo qualche secondo sarà abilitato il tasto scrivi.

A questo punto possiamo scrivere il nostro programma sul PLC. Premere appunto il tasto scrivi. Il programma passa dal personal computer al PLC. Le barre di posizione si incrementano per tutta la lunghezza del

programma assicurando che il programma scritto sia esatto attraverso una verifica nei messaggi inviati.

Una volta che il programma è stato passato sul plc selezionare la cartella **Stato Plc**.

Compare la seguente finestra:

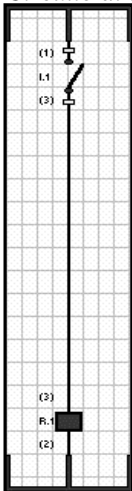


Premere il tasto Run per far partire l'esecuzione del programma.

Sul display del PLC compare la scritta STOP e anche sul display virtuale compare la scritta stop. Il Fronte F.2 è attivo. Attivando l'ingresso I.1 compare la scritta START sia sul display del PLC che nel display virtuale. O.1 si attiva con I.1 e si disattiva attivando I.1. **Il programma funziona Correttamente**

Passiamo ad analizzare i blocchi fondamentali che compongono uno schema elettrico. Generalmente è sempre una combinazione di questi blocchi a comporre lo schema.

Circuito di appoggio:



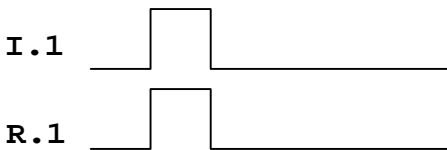
Copia lo stato di un contatto in una bobina di qualsiasi tipo.

Volendo assimilare questo schema con logiche TTL lo potremmo equiparare ad un buffer.

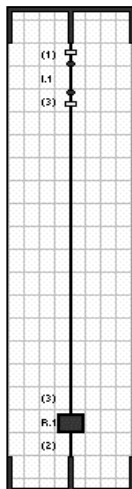
Se I.1 è attivo R.1 è attivo.

Se I.1 è disattivo R.1 è disattivo

Lo schema temporale di questo circuito è il seguente:



Il circuito che ha funzionamento opposto è il seguente:



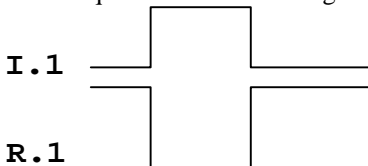
Copia lo stato invertito di un contatto in una bobina di qualsiasi tipo.

Volendo assimilare questo schema con logiche TTL lo potremmo equiparare ad un NOT.

Se I.1 è attivo R.1 è disattivo.

Se I.1 è disattivo R.1 è attivo

Lo schema temporale di questo circuito è il seguente:



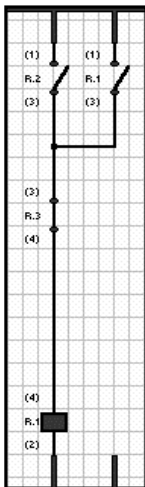
Il prossimo circuito ,già visto nell'esempio di apertura e il circuito di memoria di uno stato:

Questo circuito può essere assimilato ad un flip flop Set-Reset.

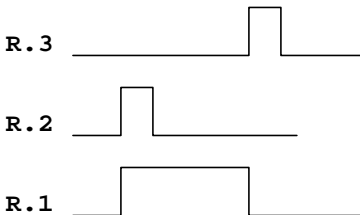
Nell'esempio specifico R.2 è l'ingresso di Set,R.3 è l'ingresso di Reset R.1 è l'uscita.

La tavola della verità è la seguente

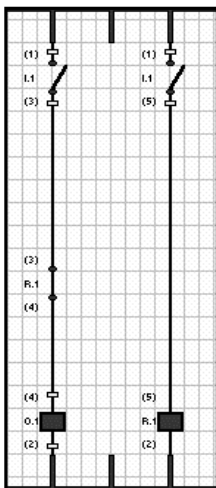
R.2	R.3	R.1
0	0	N-1
1	0	1
0	1	0



Mentre il seguente diagramma ne descrive il comportamento temporale:



Il seguente circuito implementa un rilevatore di fronte eseguito con componenti base e può risultare utile ogni volta che siano stati usati tutti e 32 i fronti a disposizione:



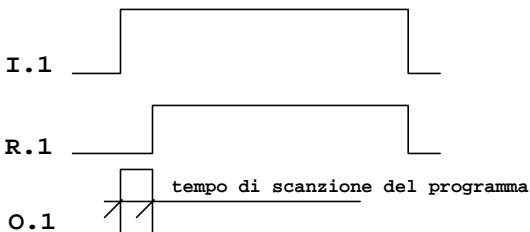
Per spiegare il funzionamento di questo circuito bisogna introdurre il concetto di scansione programma. Uno schema elettrico o comunque qualsiasi programma inizia alla istruzione START e termina all'istruzione END. Questo ciclo identifica la scansione del programma, che quindi ha un tempo per quanto piccolo, finito. Dopo l'istruzione END il sistema operativo del PLC esegue vari controlli sul funzionamento del sistema e riavvia una nuova scansione. Bisogna però assimilare questo concetto:

Nella istruzione START vengono letti gli stati attuali degli ingressi hardware. Nella istruzione END vengono scritte le uscite Hardware del PLC.

Uno schema elettrico viene eseguito dal PLC spostandosi da sinistra in alto, prima verso il basso e poi verso destra.

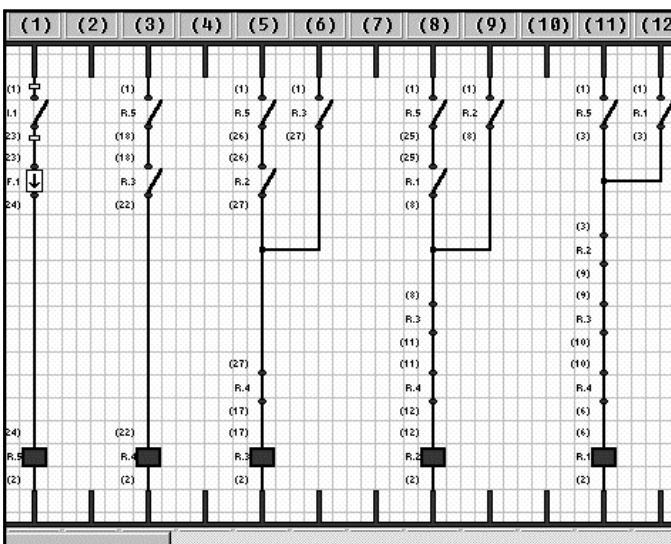
Allora nello schema di figura, poiché gli ingressi vengono aggiornati solo nella istruzione START, lo stato di I.1 è lo

stesso per ambedue le colonne anche se la lettura viene eseguita in due istanti diversi. Quindi quando si attiva I.1 la scansione di programma processa la prima colonna il cui risultato è quello di attivare O.1, poi processa la seconda colonna il cui risultato è quello di attivare R.1. Il sistema rimane in questo stato fino alla prossima scansione di programma in quanto questa volta mentre la seconda colonna dà come risultato R.1 attivo la prima dà come risultato O.1 disattivo. Il sistema rimane in questo stato fino a che non viene rilasciato I.1.



Applicando il concetto precedente e il circuito di memoria già visto si può implementare un sequenziatore di stati .

Prendiamo in esame il seguente circuito:



Nella colonna 1 viene eseguito un rilevatore di fronte sul rele R.5,quindi R.5 è attivo per un solo ciclo di scansione. Alla partenza del programma R.1,2,3,4 sono tutti disattivi. Al primo impulso di R.5 si attiva R.1 e rimane attivo per ch  è un circuito autointrattenuto Con R.2,3,4 come reset.Al Secondo

impulso di R5 R.1   attivo per cui abilita il circuito di colonna 8 e si attiva R.2 ma ci  disattiva R.1 sulla colonna 11.R.2 Rimane attivo perch    su un circuito di memoria con R.5 come SET,R.1 come abilitazione del SET,mentre R3,4 come RESET.Al nuovo impuso di R5 si avra la attivazione di R.3 e la disattivazione di R.2 perch  il circuito   analogo a quello appena analizzato.

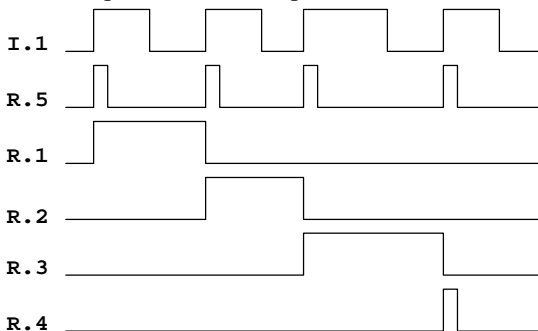
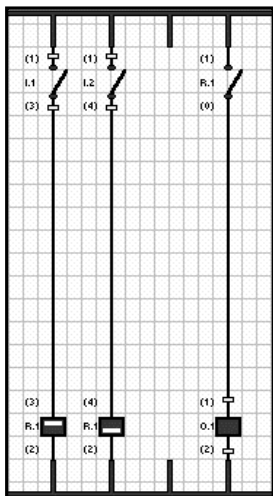
Al successivo impulso si attiva R.4 che   comune a tutte le colonne e esegue un reset assoluto di tutti gli stati di memoria,impostatndo lo stato iniziale.

Bisogna analizzare che:

Se il circuito fosse stato disegnato in ordine crescente con la prima colonna con la bobina R1 poi R2,3,4 il circuito non funziona e al primo impulso di R.5 si attiverebbe tutti i rele per il tempo di scansione.(dimostrare perch  ?).

Lo stesso circuito pu  essere ripetuto per un numero di volte tante quante ne occorrono

Ripetendo la sequenza abilitazione impulso con relè successivo e reset con rele precedente, inoltre modificando il reset si può avere la abilitazione dei vari stati senza interrompere lo stato precedente e anche questo circuito è spesso usato.



Il microPLC merlino mette a disposizione anche bobine SET e RESET impostabili su un rele quindi ,ad esempio un circuito di memoria come visto in precedenza puo essere così costituito:

Se si attiva I.1 la bobina SET di R.1 attiva R.1

Se si attiva I.2 la bobina RESET di R.1 disattiva R.1

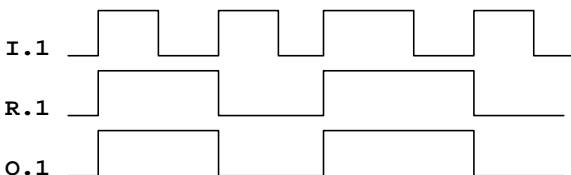
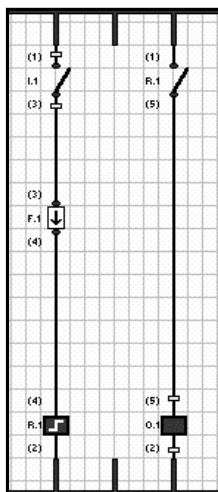
Nell'esempio di figura l'uscita O.1 si attiva premendo I.1 e si disattiva premendo I.2

Si ha inoltre a disposizione anche bobine monostabili (saltarello) nel senso che il relè interessato cambia di stato ogni volta che si attiva questa bobina.

Una simile bobina per dare un risultato stabile deve per forza avere un rilevatore di fronte altrimenti il risultato è dato dal tempo di attivazione del contatto diviso il tempo di scansione del programma ,se il risultato è pari la bobina è spenta ,se è dispari e attiva(perché ?).

Lo schema con la bobina monostabile è il seguente:

Il diagramma temporale di questo schema è il seguente:

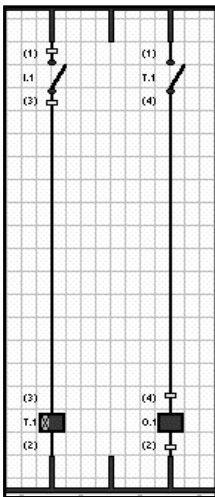


Prendiamo ora in esame i circuiti con i timer.

Con timer definiamo una bobina che quando attivata chiude i suoi contatti dopo un certo tempo in precedenza impostato.

Questo tipo di timer viene definito timer ritardato alla eccitazione. E' comunque possibili avere i tipi di timer ma è lasciato al programmatore la implementazione.

Il primo circuito da disegnare è il seguente:



Le bobine dei timer sono identificate dal seguente bottone:



Una volta disegnato lo schema di figura posizionare il mouse sopra alla bobina del timer e inviare un doppio click con il tasto sinistro del mouse. Compare il seguente dialog di impostazione timer:

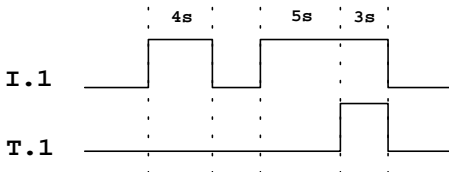
The dialog box has a title bar 'Numerazione Fili (Timer)'. It is divided into two main sections: 'Collegamenti' and 'Setting'.
 In the 'Collegamenti' section:
 - 'Up=' is set to 3.
 - 'Down=' is set to 2.
 - 'Nome =' is set to 'TIMER'.
 - 'Num. =' is set to 1.
 - 'Comm =' is an empty text field.
 In the 'Setting' section:
 - There are radio buttons for 's./10' (selected), 'min.', 'sec.', and 'ore'.
 - 'Set =' is set to 50.
 At the bottom, there are 'Cancel' and 'OK' buttons.

impostando nel campo Setting i dati di figura si setta il timer a 50 decimi di secondo. Impostare 5 secondi anche se

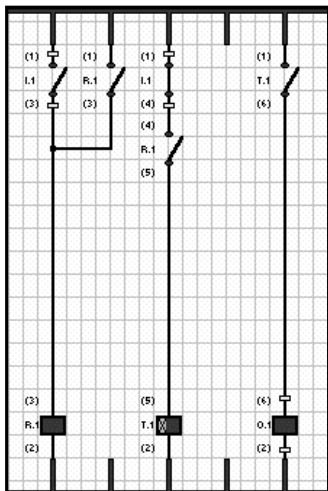
aprima vista può sembrare identico ,non lo è perché l'errore è pari a una unità della scala impostata quindi nel primo caso si può avere tempi tra 4,9 e 5,1 secondi mentre nel secondo caso tra 4 e 5 secondi. Analizziamo lo schema.

Se I.1 è attivo la bobina del timer viene abilitata e il timer inizia a contare il tempo trascorso, in questa fase il contatto del timer T.1 è ancora aperto perché deve trascorrere tutto il tempo impostato. Se si disattiva l'ingresso I.1 prima che sia trascorso tutto il tempo del timer T.1 il timer viene resettato e non succede nulla. Invece se I.1 è attivo per tutto il tempo di T.1 trascorso questo

tempo il contatto di T.1 si chiude e si attivera O.I. Il diagramma temporale è il seguente:



Per ottenere un timer ritardato al rilascio disegnare il seguente schema:



Come si vede si fa ricorso ad un circuito di memoria su R.1. La prima colonna è il circuito di memoria, la seconda rileva il rilascio del contatto I.1 e attiva T.1. Al momento di T.1. Avremo quindi il seguente diagramma temporale:

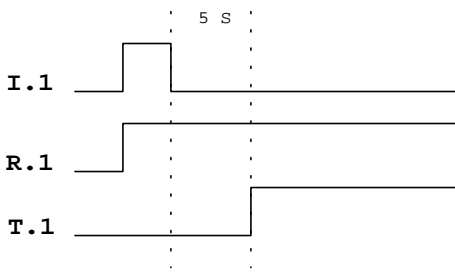
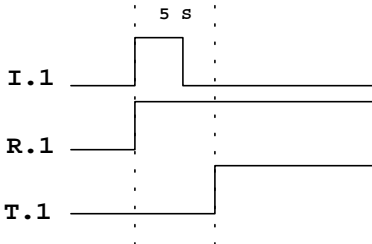
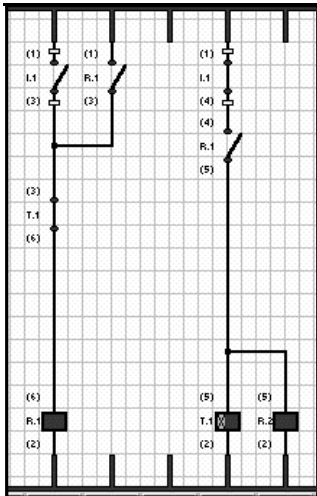


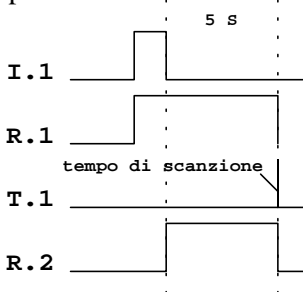
diagramma temporale:

Il circuito di figura così come è può essere usato una sola volta perché manca il circuito di reset su R.1. Per ottenere un impulso di T1 pari al tempo di scansione si può immettere sulla prima colonna un contatto N.C. di T.1. oppure si può fare resettare il circuito dalle bobine comandate da T.1. Se si voleva iniziare il conteggio del timer sul fronte di salita dell'impulso I.1 basta togliere il contatto N.C di I.1 sulla prima colonna. In questo caso il diagramma temporale diviene:

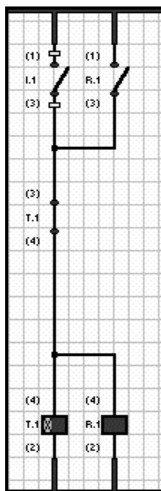




Infine si puo' avere un impulso di durata prestabilita al rilascio con lo schema a fianco: l'uscita dell'impulso e' su R.2 :

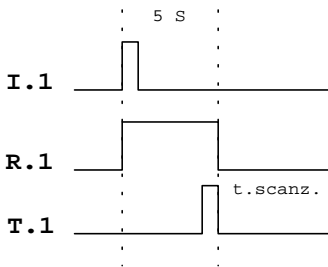


Per ottenere un timer monostabile disegnare il seguente schema:

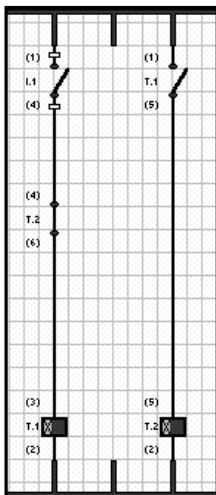


Un qualsiasi impulso di I.1 viene catturato dal circuito di memoria su R.1. Ma lo stesso circuito attiva T.1 che resetta R.1 trascorso il proprio tempo di impostazione. L'uscita è un contatto di R.1.

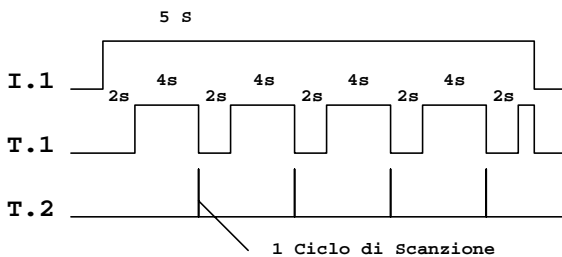
Si ha il seguente diagramma temporale:



Per ottenere un timer ciclico bisogna ricorrere al seguente schema:

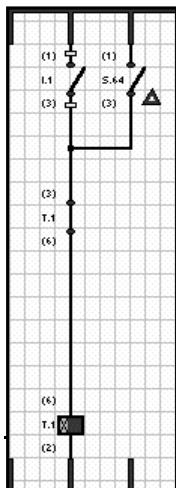


I.1 abilita T.1 ,trascorso il tempo impostato in T.1 viene abilitato T.2 ,trascorso il tempo di impostazione di T.2 viene resettato T.1 e inizia un nuovo ciclo, fintato che I.1 e' attivo. Avremo quindi una sequenza di impulsi di tempo pari alle due impostazioni dei timer. Supponedo T.1 impostato a 20 decimi di secondo e T.2 impostato a 40 decimi di secondo avremmo i seguenti stati:



Seppure i blocchi precedenti cosi' come sono stati presentati possono sembrare complessi , questi possono essere semplificati assimilando il circuito di memoria attraverso i contatti istantanei dei timer. Grazie a questo contatto la bobina del timer si puo' anche comportare da rele'. I contatti istantanei si trovano impostando un contatto speciale con la funzione *contatto istantaneo del timer n*

Ad esempio lo schema del timer monostabile visto in precedenza diviene:



l'uscita e' il contatto istantaneo.

Approfondimento:

Nel manuale utente del microPLC merlino è specificato nel capitololo data base la mappatura interna dei timer.

Dal registro R(215) al registro R(238) vi sono le impostazioni del valore e lo stato del conteggio (Low Word = impostazione ,High Word = stato del conteggio)mentre dal registro R(239) al registro R(244) vi è la definizione del modo di funzionamento e lo stato del timer.

Di particolare importanza per il programmatore esperto sono i primi 24 registri specificati in quanto molto spesso serve di

reimpostare il settaggio di un timer attraverso una pagina LCD.

Per eseguire questa funzione basta quindi modificare la Word bassa dei precedenti registri:

R(215).LW	Impostazione timer 0
..	
R(238).LW	Impostazione timer 23

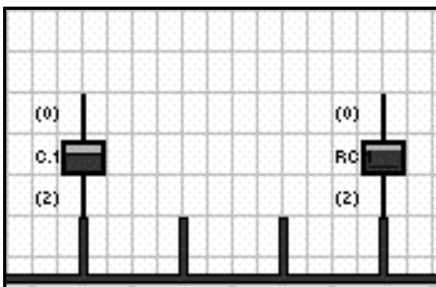
Il sistema operativo del microPLC scandisce i timer solo se è stata specificata una istruzione di impostazione dei timer SETTIMER, altrimenti si presuppone che nessun timer sia usato. In particolare se sono presenti dei timer il registro di presenza componenti R(214) imposta i primi 4 bit se:

BIT 0	Vi sono dei timer a 1/10 sec
BIT 1	Vi sono dei timer a 1 sec
BIT 2	Vi sono dei timer a minuti
BIT 3	Vi sono dei timer a ore

Se manualmente si resetta questi bit i timer con la impostazione corrispondente rimangono bloccati.

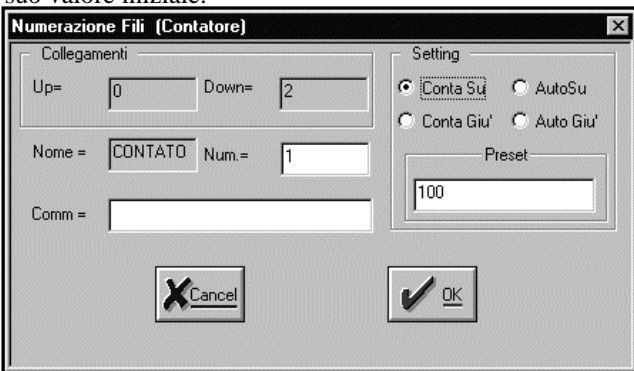
Prendiamo ora in esame i circuiti con i contatori

Ogni PLC contiene al suo interno **24 contatori**. I contatori sono programmabili nel range compreso tra 1 e 32768, ed in modalità UP e DOWN. Come per i relè interni ed in genere come per tutti i componenti del microPLC Merlino, ogni *CONTATORE* può



disporre di un numero illimitato di contatti aperti e/o chiusi (sempre in rapporto alla capacità di memoria di programma). Il simbolo che contraddistingue il contatore è quello rappresentato nel disegno a fianco e contrassegnato dal nome **C1**. All'interno del PLC i contatori vengono emulati dal software, in tempo reale e rispecchiando il funzionamento dei contaimpulsivi tradizionali. Il numero di identificazione, ed

il valore di PRESET con la relativa modalità di conteggio, possono essere impostati cliccando due volte all'interno del rettangolo che rappresenta il contatore. Più propriamente il rettangolo raffigurato e denominato **C1** è solo la bobina del modulo di conteggio. Come per i **TIMER** tradizionali anche questo componente è costituito nel suo insieme dalla bobina e dai contatti **NA**, **NC**. Il contatore potrà essere resettato per mezzo di una bobina di reset **RC1**, la quale una volta abilitata riporta il conteggio al suo valore iniziale.



La finestra di dialogo per la impostazione dei valori di SET è raffigurata a fianco; è ben evidente che cliccando all'interno di uno dei 4 cerchietti è possibile selezionare la modalità di conteggio, mentre dentro alla finestra **PRESET** è possibile inserire il

valore di preset del contatore. Come per gli altri componenti visti in precedenza è possibile immettere il numero di identificazione (all'interno della finestra **NUM**) ed un eventuale commento allo scopo di facilitare la lettura del programma.

Il funzionamento del contaimpulsivo è molto semplice e dipende dalla modalità impostata.

Modalità *CONTASU*

- Il contatore inizia il suo conteggio da 0.
- Ogni volta che la bobina **C1** viene abilitata il contatore si incrementa di 1. La funzione di antirimbato è già supportata internamente dal software e

per il corretto conteggio degli impulsi non è necessario inserire nessun fronte di salita.

- Quando il conteggio ha il raggiunto il valore di PRESET – 1 i contatti normalmente aperti si chiudono e quelli chiusi si aprono.
- Al raggiungimento del valore prefissato, se la bobina non viene interrotta, il conteggio continua sino al valore massimo (32768) per poi ritornare a 0, e quindi, la perdita dello stato attuale. Per ovviare a questo inconveniente basta inserire in serie alla bobina di conteggio un contatto N.C. del contatore stesso
- I contatti aperti e/o chiusi ritornano nel suo stato iniziale solo dopo che il contatore è stato resettato per mezzo della bobina di reset.

Modalità CONTAGIU

- Il contatore inizia il suo conteggio dal valore di preset
- Ogni volta che la bobina di C1 viene abilitata il contatore si decrementa di 1. La funzione di antirimbalzo è già supportata internamente dal software e per il corretto conteggio degli impulsi non è necessario inserire nessun fronte di salita.
- Quando il conteggio ha il raggiunto il valore di 0 i contatti normalmente aperti si chiudono e quelli chiusi si aprono.
- Al raggiungimento del valore prefissato, se la bobina non viene interrotta, il conteggio continua sino al valore massimo (32768) per poi ritornare a 0, e quindi, la perdita dello stato attuale. Per ovviare a questo inconveniente basta inserire in serie alla bobina di conteggio un contatto N.C. del contatore stesso
- I contatti aperti e/o chiusi ritornano nel suo stato iniziale solo dopo che il contatore è stato resettato per mezzo della bobina di reset.

Modalità AUTOSU

- In questa modalità il contatore funziona come un timer veloce con una base dei tempi pari a 20 ms
- Il contatore inizia il suo conteggio da 0.
- Il contatore fintanto che è abilitato si autoincrementa per una unità ogni 20 ms.
- Quando il conteggio ha il raggiunto il valore di PRESET – 1 i contatti normalmente aperti si chiudono e quelli chiusi si aprono.
- Al raggiungimento del valore prefissato, se la bobina non viene interrotta, il conteggio continua sino al valore massimo (32768) per poi ritornare a 0, e quindi,

la perdita dello stato attuale. Per ovviare a questo inconveniente basta inserire

in serie alla bobina di conteggio un contatto N.C. del contatore stesso

- I contatti aperti e/o chiusi ritornano nel suo stato iniziale solo dopo che il contatore è stato resettato per mezzo della bobina di reset.

Modalità AUTOGIU

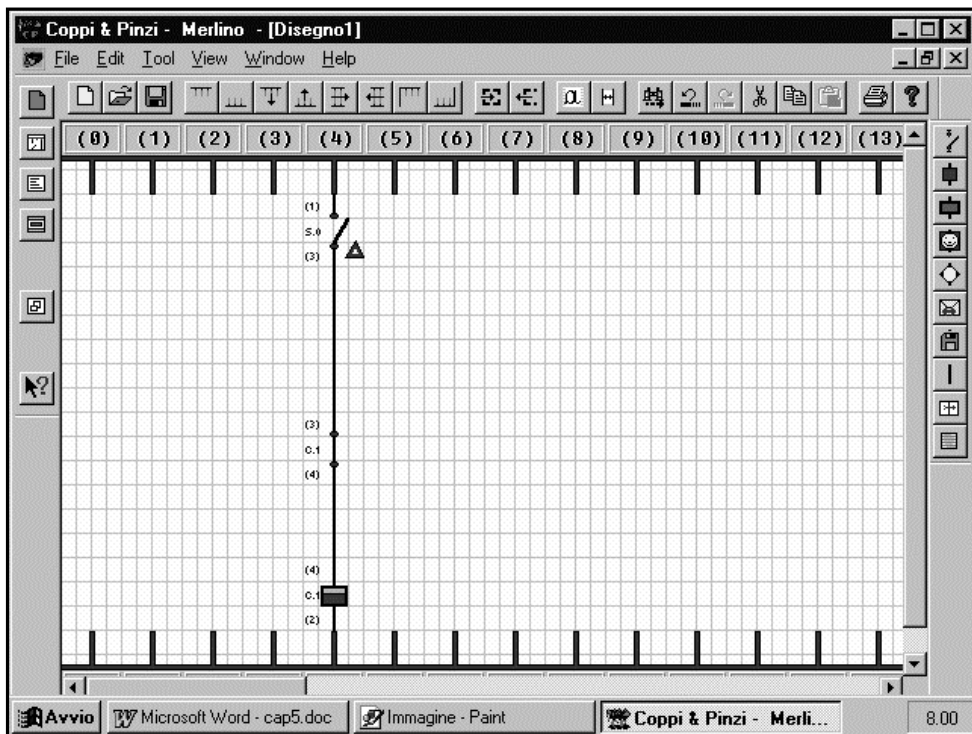
- In questa modalità il contatore funziona come un timer veloce con una base dei tempi pari a 20 ms
- Il contatore inizia il suo conteggio dal valore di preset
- Il contatore fintanto che è abilitato si autoincrementa per una unità ogni 20 ms.
- Quando il conteggio ha il raggiunto il valore di 0 i contatti normalmente aperti si chiudono e quelli chiusi si aprono.
- Al raggiungimento del valore prefissato, se la bobina non viene interrotta, il conteggio continua sino al valore massimo (32768) per poi ritornare a 0, e quindi,
la perdita dello stato attuale. Per ovviare a questo inconveniente basta inserire
in serie alla bobina di conteggio un contatto N.C. del contatore stesso
- I contatti aperti e/o chiusi ritornano nel suo stato iniziale solo dopo che il contatore è stato resettato per mezzo della bobina di reset.

Contatori come timer da programma

In modalità autosu e autogiu i contatori funzionano come timer veloci e se richiesti funzionano perfettamente allo scopo. Si può comunque impostare un contatore come timer da programma facendo contare allo stesso modo un contatto speciale a tempo ad esempio ad 1/10 di secondo, la differenza essenziale tra le due configurazioni sta nel tempo assorbito dalle due funzioni al sistema. La prima funzione da questo punto di vista è senz'altro più pesante in quanto implementa ad interrupt il refresh dei contatori indipendentemente dal tempo di scansione, mentre la seconda risulta essere una normale applicazione di un contatore, quindi se non richiesto in modo particolare è più conveniente usare quest'ultimo modo per espandere i 24 timer già presenti nel sistema

I contatti speciali a tempo presenti nel microPLC sono 10, S(0..9).

Esempio Contatore come timer:



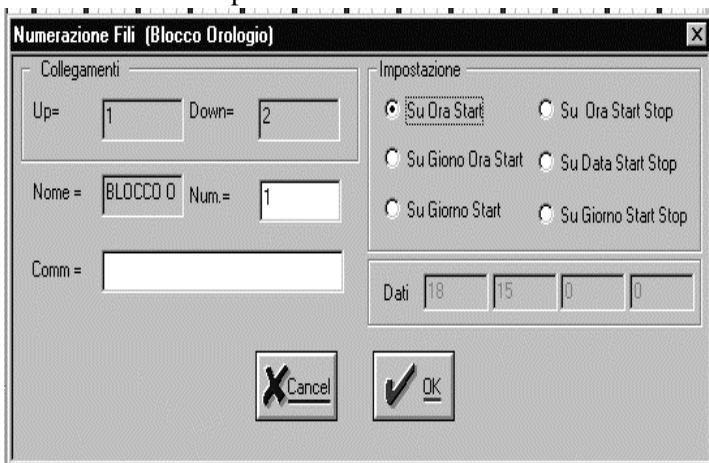
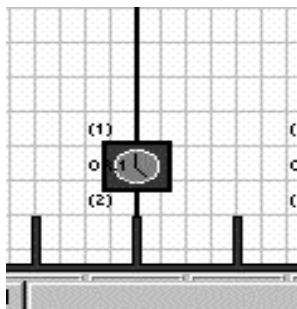
Approfondimento

Il sistema operativo mappa i contatori dal registro R(245) al registro R(274). L'impostazione e lo stato del conteggio va per i 24 contatori dal registro R(245) al registro R(268), dove la parte bassa del registro è la impostazione del contatore (Low Word= impostazione) mentre lo stato del conteggio è la parte alta(High Word=conteggio).

Il modo di funzionamento del contatore viene impostato nei registri che vanno da R(269) a R(274).

Se viene usato un contatore come timer veloce viene impostato il BIT 4 del registro presenza componenti R(214).

Il PLC Merlino è dotato di un orologio interno che mantiene memorizzati data ed ora in tempo reale anche in assenza di alimentazione. Utilizzando le funzioni dell'orologio interno sono stati implementati 8 blocchi bobina orologio che offrono notevoli possibilità di programmazione. Tutti i blocchi sono perfettamente identici ed ognuno possiede un numero illimitato di contatti NA ed NC. Ognuno di questi orologi è impostabile in diverse modalità come è possibile vedere dal disegno raffigurante la finestra di dialogo che, naturalmente, è accessibile cliccando due volte con il pulsante sinistro del mouse sulla relativa bobina.



Dal dialog box, all'interno del riquadro impostazione, sono visibili i modi di configurazione del modulo orologio.

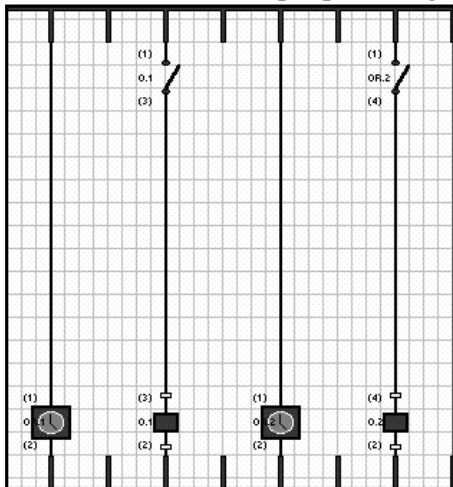
- Su ora start impostando l'ora ed i minuti all'interno del riquadro DATI stabiliremo il momento in cui i contatti NA dell'orologio definito da "NUM" (1,2,3 o 4) si chiudono, e viceversa, quelli NC si aprono. In questa modalità i contatti utilizzati restano attivi per un solo minuto, questo perché i dati dell'orologio sono aggiornati con cadenza al minuto.
- Su ora start stop in questa modalità va definito il momento di start inserendo l'ora ed i minuti nelle prime due caselle sulla sinistra (all'interno della finestra dati), e l'ora ed i minuti di stop nelle caselle

rimanenti. In questo modo il funzionamento è analogo a quello di un orologio giornaliero a cavalieri.

- Su Giorno Ora Start il funzionamento è quello di un orologio annuale. Vanno definiti: ore, minuti, giorno del mese e mese dell'anno. Anche in questo caso i contatti restano attivi per il solo minuto corrente nel quale avviene lo start.
- Su Data Start Stop in questo modo l'orologio rimarrà attivo per l'intervallo stabilito dai giorni e mesi di start e di stop definiti nelle finestre DATI. In questo caso la base dei tempi è scandita dal passaggio per la mezzanotte del giorno impostato.
- Su Giorno Start In questo caso ci riferisce al giorno settimanale che è un numero compreso tra 1 e 7. Anche in questo caso lo start avviene al passaggio delle ore 24.
- Su Giorno Start Stop come la selezione precedente ma i bit interni dell'orologio selezionato vengono abilitati nell'intervallo dei giorni settimanali stabiliti.

Come per tutti gli altri blocchi funzione è possibile inserire un commento per identificare qual è la funzione svolta dall'orologio inserito; inoltre è necessario inserire il numero che identifica l'orologio e che per questo tipo di PLC va da 1 a 8

In questo esempio vengono utilizzati tre moduli "orologio" impostati in modi diversi l'uno dall'altro per poter meglio comprenderne il funzionamento.



DESCRIZIONE :

Lo schema è molto semplice; l'orologio N°1 il cui contatto NA abilita la uscita N°1 (colonne 1 e 3) è impostato in modalità *ORA START* per le ore 18:15. Raggiunta l'ora prestabilita abiliterà l'uscita N°1 per un solo minuto. Naturalmente nel caso che occorra prolungare il tempo di abilitazione si può ricorrere alla funzione *ORA START STOP*. Questo è il modo con cui è stato programmato il blocco orologio N°2. Operando un doppio click su questa bobina è possibile verificare o reimpostare

i tempi di lavoro. L'uscita interessata dall'orologio N°2 è l'uscita N°2, anche questa uscita raggiunto il tempo prefissato (sempre all'interno del minuto corrente) si abiliterà accendendo il led ed il relè interessati.

I contatti tasto sono la prima particolarità del microPLC merlino rispetto al microPLC D1208, infatti gli esempi fino ad ora riportati potevano essere implementati anche su tale PLC.



Un contatto tasto si chiude quando il tasto corrispondente è chiuso.

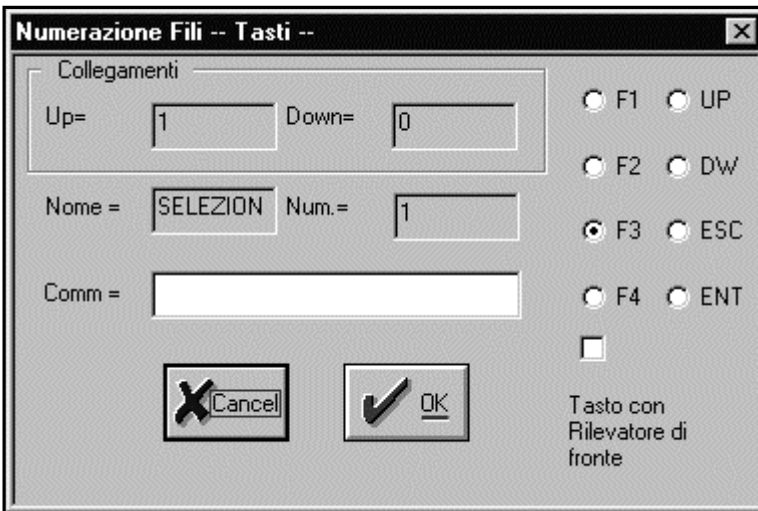
Il microPLC merlino ha 8 tasti hardware con la seguente implementazione:



Un contatto tasto non ha restrizioni e può essere usato come un qualsiasi contatto N.A.

Una volta piazzato sul disegno può essere impostato aprendo il suo

dialog box delle proprietà .Compare:

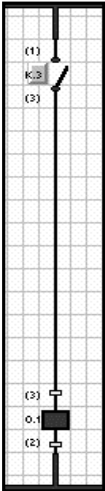


Selezionare il tasto che si vuole tra gli checkbox disponibili. Il campo Tasto con rilevatore di fronte implementa direttamente sul contatto tasto la serie con un rilevatore di fronte senza ricorrere ai

rilevatori di fronte di sistema, in modo da risparmiare risorse.

In generale per come è impostato il sistema operativo è preferibile utilizzare nei programmi i tasti F1..F4 in quanto gli altri vengono usati dalla funzione di sistema per l'aggiornamento delle variabili da utente come vedremo più avanti.

Per verificare in modo immediato il funzionamento dei tasti eseguire il seguente semplice schema:



Impostare il tasto senza rilevatore di fronte.

Quando si preme F.1 si chiude O.1.

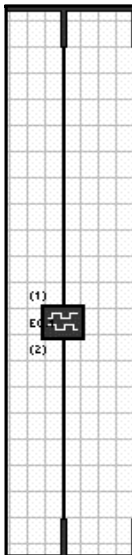
Combinazioni di tasti possono essere effettuate semplicemente ponendo in serie i vari tasti.

Approfondimento

Il sistema operativo rileva i tasti nel registro R(78).

Il primo bit del registro legge direttamente la tastiera (LLB) dando come risultato 8bit (255 tasti come combinazione degli 8).

I contatti dei tasti sono i bit memorizzati dal sistema operativo negli ultimi 8 del registro (HHB) mentre negli altri 2 byte il sistema operativo imposta i rilevatori di fronti per i tasti e la serie tra i tasti e i rilevatori di fronti.



Il microPLC ha un ingresso per encoder con segnali in quadratura. Il sistema operativo aggiorna la posizione dell'encoder ad una frequenza di 30 hz se non si imposta un modo particolare attraverso la bobina encoder ,riportata qui a fianco.L'encoder per default è impostato come assoluto e riparte da zero ad ogni accensione della macchina.

Piazzando una bobina encoder si può impostare le proprietà dell'encoder di sistema con il seguente dialog box:

Encoder: indica che in ingresso è presente un segnale in quadratura da decodificare.

Contatori: indica che in ingresso vi sono 1 o 2 segnali asincroni

Frequenza: indica a che frequenza deve essere aggiornata la posizione degli encoder o dei contatori. Impostazione 30...5000 hz. A questa frequenza vengono anche aggiornati i contatti posizione.

Assoluto: L'encoder è trattato come assoluto a 32 bit con segno

Relativo: L'encoder è trattato come relativo all'ultima lettura a 32 bit con segno

Encoder clk automatico: La frequenza per la lettura è generata esternamente da uno dei due segnali in quadratura e dipende dalla velocità di rotazione dell'encoder. Sono accettate frequenze fino a 500Khz. Il campo frequenza diviene divisore .impostare un numero tra 0 e 65535 ,indica dopo quanti impulsi viene aggiornata la posizione.

Attenzione, è lasciato al programmatore il controllo del sistema, divisori piccoli su segnali a frequenza troppo alta possono generare interrupt troppo pesanti per il sistema che si può bloccare.

Contatori su canale 0: Si assume che su il canale 0 e 1 vi siano due segnali asincroni, in questo modo si può effettuare una misura di frequenza impostando un divisore nel campo frequenza se la frequenza campione è sul canale zero dopo il divisore viene aggiornato il conteggio effettuato.

Contatori su canale 1: come su canale 0 solo che la frequenza che genera l'interrupt è sul canale 1.

Attenzione : il rilevatore di quadratura del segnale per encoder vede i fronti di salita del segnale stesso, per cui ad ogni passo vengono misurati 4 passi aumentando la risoluzione del sistema.

Approfondimento:

La gestione dell'encoder interessa vari registri tra cui:

R(275)	Posizione dell'encoder
R(276)	Posizione contatore 0 o encoder relativo se encoder impostato come assoluto
R(277)	Posizione contatore 1
R(278)	Impostazione dell'encoder

Inoltre il BIT 7 di R(214) presenza componenti viene impostato.

La frequenza di aggiornamento dell'encoder **non è** la massima frequenza che l'encoder accetta la ma frequenza con cui la posizione letta viene scritta nella memoria R(275). Per calcolare la massima frequenza in ingresso bisogna tenere conto che: Il registro hardware di lettura dell'encoder è a 16 bit e quindi anche a 30 hz di frequenza di aggiornamento occorre una frequenza pari a :

$$F_{\max}(30\text{hz})=65535*30=1.996 \text{ Mhz}$$

Prima di perdere impulsi di posizione.

Poiche l'ingresso per encoder viene pilotato da un circuito darlington al suo ingresso questo ne limita la frequenza a 500Khz.

Altro discorso vale per il controllo di posizione ,i contatti posizione sono effettuati entro l'interrupt di aggiornamento per cui con un segnale a 500Khz e frequenza di aggiornamento a 30 hz la posizione minima letta per rate è pari a

$$N.\text{Imp.}=500000/30=16666$$

Per cui il sistema operativo vede tra un rate e l'altro salti da 16666 passi. Questo è senz'altro inaccettabile in caso di posizionamento.

Assumendo esempi reali , supponiamo un motore che gira a 3000 rpm, ovvero

$$\text{Giri secondo}=3000/60=50 \text{ g/s}$$

Con un encoder a 256 step giro si ha una frequenza di lettura pari a

$$F.\text{enc}=50*(256 *4)=51200.$$

Con una frequenza di lettura a 5Khz si ha una risoluzione di posizionamento pari a 10 fronti e 2,5 step.

$$\text{Fronti } =51200/5000=10.24$$



I contatti posizione sono dei contatti N.A. che si chiudono quando la posizione dell'encoder ha raggiunto un determinato valore.

Questo tipo di posizionamento non è affidabile come implementazioni attraverso funzioni PID ma è stato inserito nel sistema per applicazioni a basso livello. Qualora si renda necessario eseguire posizionamenti più precisi si può implementare in linguaggio assembler un algoritmo PID.

Quando si imposta le proprietà del contatto posizione compare il seguente dialog box:

Numerazione Fili -- Posizionatori--

Collegamenti

Up= 1 Down= 6

Nome = POSIZION Num. = 4

Comm =

Impostazioni

Maggiore e Uguale

Minore

Posizione = 0

Cancel OK

Se si imposta il campo maggiore o uguale il contatto funziona da N.A. mentre se si imposta minore il contatto funziona come N.C.

Nel campo posizione si immette il numero a 32 con segno su cui effettuare il confronto. Non vi è nessuna restrizione all'uso del contatti posizione.

Approfondimento:

I contatti posizione sono elaborati nell'interrupt ,poi assimilati e elaborati nel programma utente .

Le posizioni associate ad ogni contatto sono memorizzate nei registri R(279) .. R(286)

Mentre il BIT 7 del registro R(214) presenza componenti è impostato a 1.



I contatti test permettono di effettuare test comparativi tra tutti i registri del sistema operativo .

Il contatto test è un contatto N.A. che si chiude quando la condizione su cui è impostato è vera.

Le proprietà di un contatto test sono impostabili tramite il seguente dialog box:

I 6 tasti in basso impostano la condizione da verificare e sono rispettivamente: **UGUALE, DIVERSO, MAGGIORE O UGUALE, MAGGIORE, MINORE, MINORE O UGUALE**

Nel campo Par.1 si può selezionare se il test è effettuato tra registri o con valori numerici.

Il listBox centrale permette la selezione tra i 7 formati possibili su cui eseguire il test mentre gli edit box laterali permettono di impostare l'indirizzo del registro su cui effettuare la condizione.

Ad ogni variazione di questi campi viene rielaborata la equazione attuale e visualizzata nel primo campo in alto.

Ad esempio per avere una soglia analogica sull' ingresso analogico 1 (R64).LW basta impostare l'espressione $R(64).LW > \#500$ e il contatto test corrispondente si chiude quando il valore analogico sull'ingresso 1 supera il valore 500.

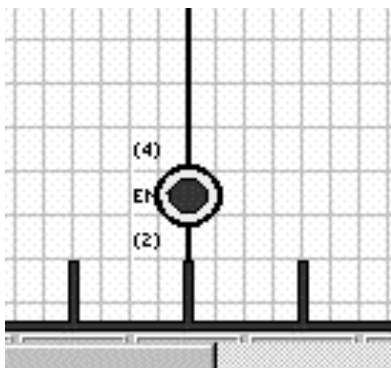
I contatti test non hanno restrizioni particolari e possono essere usati in serie e parallelo a piacimento.

Approfondimento:

Non vi sono particolari restrizioni all'uso dei contatti test in un programma, ma è meglio non eccedere perché ogni contatto viene implementato dal compilatore con diversi passi di programma.

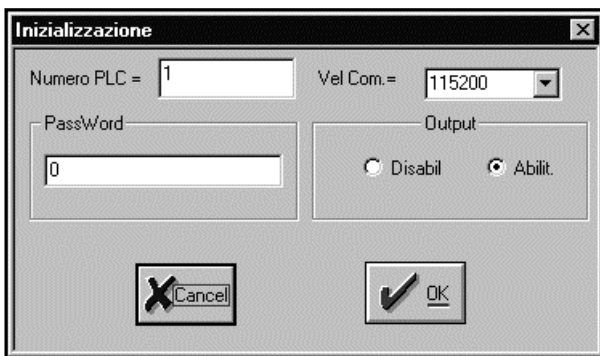
Se nello schema sono presenti contatti test con lo stesso numero il compilatore genera degli errori.

Quello raffigurato sotto è il componente ENABLE. La funzione di questo blocco è quella di forzare lo stato di tutte le uscite a zero quando non è abilitato. Questo blocco può essere utilizzato quando vogliamo essere certi che, a prescindere dallo stato di ciascuna uscita, un unico comando ne forzi la condizione allo 0 logico.



disabilitati.

- *PASSWORD=imposta un numero do password*

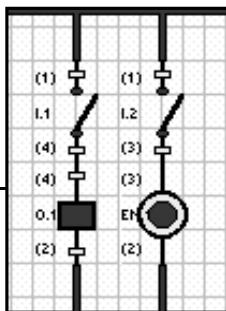


E' importante ricordare che la modifica di questi parametri non viene salvata nel disegno, per cui ad ogni nuova riapertura dello stesso questi vanno nuovamente settati nella maniera voluta.

Il blocco ENABLE è l'unico parametro di inizializzazione che può essere cambiato anche da schema elettrico, e questo è possibile per mezzo della bobina descritta in precedenza. Per comprendere meglio questa funzione abbiamo realizzato un piccolo esempio qui a fianco riportato.

Inserendo l'ingresso I.1 l'uscita O.1 si attiva ma il relè fisico del PLC non si chiude fino a che non si attiva I.2 che abilita la scrittura delle uscite hardware del PLC.

Approfondimento:



La abilitazione delle Uscite dipende dallo stato del registro R(6).quando è a 1 le uscite sono abilitate altrimenti no.

La abilitazione delle uscite agisce anche sulla scrittura delle uscite a eventuali moduli slave.



La bobina uscita analogica imposta l'ingresso analogico 6 e 7 come uscita ad un valore prefissato. Vedere il manuale utente per impostare i ponti interni al PLC.

Le uscite analogiche sono a 8 bit e accettano valori da 0 a 255.

Le proprietà di questa bobina sono impostabili tramite il seguente dialog box:

Canale 1,2=impostare quale uscita analogica si vuole modificare

Valore =impostare il nuovo valore di uscita.

Le uscite analogiche sono impostate dal sistema operativo entro l'interrupt di lettura dell'encoder questo per poterle modificare alla massima frequenza su controlli di posizione ad esempio per variare la velocità ad un inverter.

La tensione di uscita va da 0 a 5 volt.

Approfondimento:

Il valore dell'uscita analogica è memorizzato nel registro R(68).LW per il DAC 0 e R(68).HW per il DAC 1.

Una qualsiasi impostazione di questo registro varia le uscite analogiche.

L'istruzione SETDAC associata ad ogni bobina uscita analogica informa il sistema operativo che il corrispondente canale analogico (6 per DAC 0 e 7 per DAC 1) sono delle uscite analogiche.

Alla partenza del sistema questi due canali sono ingressi analogici.



La bobina calcolo una tra i componenti più complessi che il sistema mette a disposizione.

Questa bobina permette di impostare o copiare parti o per intero i vari registri di sistema. Per impostare le proprietà della bobina accedere al seguente dialog box:

I 17 tasti in basso impostano le seguenti operazioni:

IMPOSTA, PIU', MENO, MOLTIPLICATO, DIVISO, NEGAZIONE, COMPLEMENTO A 2, SHIFT A DESTRA, SHIFT A SINISTRA, ROTAZIONE A DESTRA, ROTAZIONE A SINISTRA, INCREMENTO DI 1, 2, 4, DECREMENTO DI 1, 2, 4

Nel campo Par.1 selezionare se l'operazione ha come operando un numero o un contenuto di un registro.

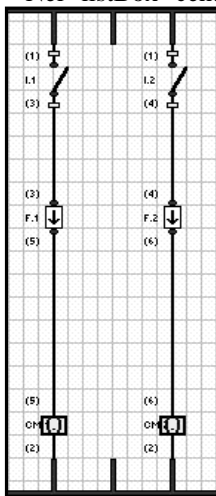
Nel listBox centrale selezionare tra i 7 modi la parte del registro interessata dall'operazione mentre negli edit box laterali impostare gli indirizzi dei registri o il valore numerico.

Ad ogni cambiamento delle impostazioni viene rielaborata e scritta la espressione corrispondente ai dati selezionati e visualizzata nel campo in alto come in figura.

Ad esempio lo schema a fianco implementa un contatore su (I.1 R(300).L= R(300).L+#1) e giu (I.2 R(300).L= R(300).L-#1). Per avere l'uscita del conteggio ad un valore predeterminato impostare un contatto test su R(300)

Approfondimento:

Non vi sono restrizioni all'uso delle bobine calcolo. bisogna assicurarsi che :Non vi siano due bobine con lo stesso numero perché altrimenti il compilatore genera errori e non eccederne nell'uso poiché ogni bobina viene tradotta in un insieme di diverse istruzioni.





I contatti generici sono contatti generali che invece di essere associati ad una particolare memoria possono essere impostati direttamente dall'utente a quali memoria fare riferimento.

Questo può essere utile ad esempio per formattare in modo particolare registri per messaggi ad un personal computer oppure per operazioni immediate .

Analogamente ai contatti (operazioni di lettura) sono supportate anche le bobine generiche:



questi componenti sono associati ad operazioni di scrittura.

Questi due tipi di componenti hanno le stesse proprietà che possono essere impostate con il seguente dialog box:

Numerazione Fili -- Contatti o Bobine Generici

Collegamenti

Up= Down=

R=

Nome = Num. =

Bit=

Comm =

Nel campo R= si può immettere l'indirizzo 0...1000 interessato da questa operazione

Nel campo Bit si può immettere il numero di Bit 0..31 interessato da questa operazione

Non vi sono particolari restrizioni all'uso dei componenti generici

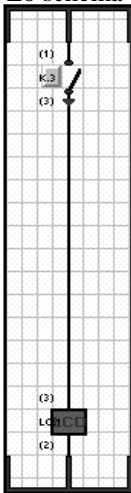
Le bobine LCD sono dei componenti che impostano dei messaggi sul display LCD.



Il numero massimo di blocchi è dato dalla sola quantità di memoria del microPLC e in generale mediamente per pagine con poche variabili viene occupato 1K di memoria ogni 15-20 pagine.

La bobina LCD negli schemi va utilizzata come una qualsiasi, quando è abilitata la pagina associata a questa bobina viene scritta sul display LCD. Poiché questa operazione viene effettuata ad ogni scansione di programma fintanto che l'abilitazione è presente conviene impostare un rilevatore di fronte altrimenti si nota un leggero tremolio sul display dovuto alla continua riscrittura della pagina. Supponiamo che vogliamo impostare una pagina che una volta visualizzata tramite il tasto F1, faccia vedere lo stato di impostazione del timer 1.

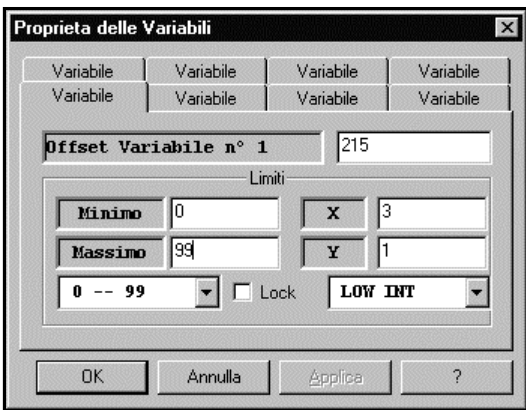
Lo schema è il seguente:



Il tasto F1 ha già in serie il rilevatore di fronte.



Per impostare la pagina andare alle proprietà dell'oggetto. Compare il seguente dialog Box:



Nel display virtuale del dialog cliccare con il mouse e scrive il messaggio di figura.

Premere quindi il bottone variabili. Compare il dialog box a fianco.

<i>Offset variabile</i>	impostare l'indirizzo della memoria a cui si vuole fare riferimento(nell'esempio R(215))
<i>Minimo</i>	valore minimo ammesso per la variabile
<i>Massimo</i>	valore massimo ammesso per la variabile
<i>X</i>	Colonna sul display 0...15
<i>Y</i>	Riga sul display 0...1
<i>Lock</i>	La variabile può essere modificata o è di sola lettura
<i>List box a sinistra</i>	Selezionare formato di visualizzazione tra i vari possibili
<i>List box a destra</i>	Selezionare formato variabile

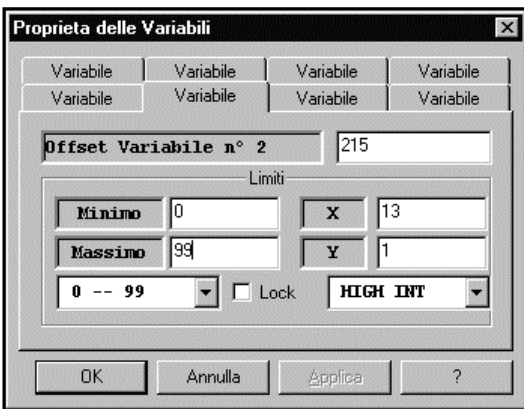
Nel dialog box viene presa in considerazione la parte bassa (LOW INT) del registro R(215) che corrisponde alla impostazione del timer 1.I parametri ammessi sono da 0 a 99 e la variabile viene scritta alla posizione XY(3,1).

Inviando Okey il precedente dialog si chiude mentre il dialog delle proprietà della bobina LCD diviene.



Adesso bisogna impostare la seconda variabile per vedere la posizione attuale del conteggio del timer.Premere il tasto Variabili.Ricompare il dialog di impostazione variabili ccome prima impostato.

Selezionare una nuova cartella variabile.



Adesso si è impostato la visualizzazione dello stesso registro R(215) ma la parte alta (HIGH INT) alle coordinate XY(13,1). Premere Okey.

Adesso il dialog box delle proprietà

diviene:



Adesso la pagina di visualizzazione è completa .

Compilare e mandare in esecuzione il precedente programma per vedere che il PLC esegue correttamente le funzioni richieste.

Approfondimento:

la visualizzazione di una pagina utente viene effettuata con la chiamata della funzione di sistema VisPagina offset 13 SYS #13,R(n).

Il registro R(n) deve contenere un puntatore ,l'indirizzo in memoria , dove risiede la struttura descrittiva la pagina da visualizzare.

Ogni volta che tale funzione visualizza una pagina copia il parametro R(n) nella memoria di sistema R(206).

Questo puntatore serve alle altre funzioni di visualizzazione per effettuare ad esempio il rinfresco dei dati visualizzati nella pagina o per la modifica della pagina stessa. Impostando a zero questo registro tutte le SYS che fanno ad esso riferimento non eseguono nessun cambiamento.

Ogni scrittura sul display viene copiata nel buffer a 32 byte (2 righe x 16 caratteri) posto nei registri R(197)..R(204).

In generale la copia del display è esatta ma quando un utente ridisegna i primi 8 caratteri del display con le SYS di sistema appropriate questi non possono essere visualizzati in forma corretta.

La memoria R(205) viene anch'essa modificata da ogni scrittura e contiene la posizione attuale del cursore (solo lettura).

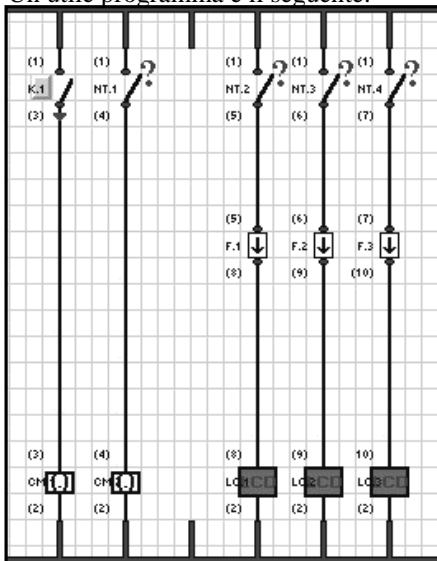
Questa bobina esegue solo la scrittura di una pagina non la modifica o il rinfresco dei dati in essa contenuti.

Se la pagina viene visualizzata in maniera continua i dati vengono rinfrescati ma si ha un leggero tremolio dovuto alle continue scritture sul display.

Anche se nel compilatore fare una pagina è molto semplice usare con parsimonia tali blocchi in quanto per ogni scrittura sul display si ha una notevole mole di dati in gioco:

Oltre ai caratteri che costituiscono la pagina è la formattazione delle variabili che può sottrarre molte risorse, appesantendo e quindi allungando il tempo di scansione. Esistono molte funzioni di sistema SYS che possono modificare l'aspetto della pagina visualizzata per avere maggiori dettagli su tali funzioni vedere il manuale utente a "funzioni di sistema", ma la visualizzazione di una pagina resetta in generale tutte queste impostazioni.

Un utile programma è il seguente:



PAGINA 2

riga 0)CONTATORI

riga 1)OROLOGI

Ad ogni pressione del tasto F3 apparirà come se il menu tendina si sposti verso il basso. Per la decodifica della pagina selezionata basta eseguire la serie con un tasto e il un test sul registro R(300).

Ad ogni pressione del tasto con rilevatore di fronte F3 il blocco calcolo esegue l'espressione $R(300)=R(300)+\#1$.

La seconda colonna con il test NT.1 controlla se R(300) è uguale a 3. Quando questa condizione è verificata il blocco calcolo CM.2 imposta $R(300)=0$.

Si ottiene che R(300) cicla sempre da 0 a 2.

NT.2 è impostato per R(300) uguale a Zero

NT.3 è impostato per R(300) uguale a Uno

NT.4 è impostato per R(300) uguale a Due.

Per cui ad ogni pressione del tasto F3 viene visualizzato le pagine da LC.1 a LC.3

Queste pagine possono essere o pagine vere e proprie (caratteri e dati) oppure solo pseudo menu ad esempio così composti:

PAGINA 1:

riga 0)TIMER

riga 1)CONTATORI

PAGINA 3

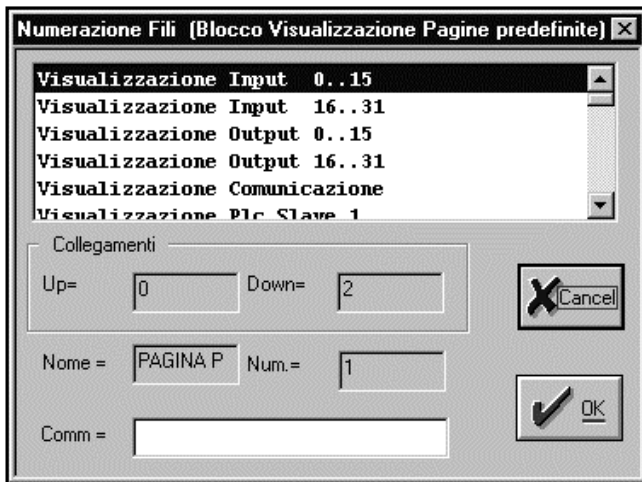
riga 0)OROLOGI

riga 1)FINE



Le bobine LCD di sistema hanno lo stesso funzionamento delle bobine LCD utente solo che la pagina è già preimpostata nel sistema operativo.

Le proprietà di queste bobine sono modificabili attraverso il seguente dialog box: Selezionare tra le varie pagine disponibili quella che più interessa.



Le pagine sono quelle che compaiono quando il plc si trova nello stato di STOP. Per il sistema operativo non vi è nessuna differenza dal punto di vista formale tra una pagina utente o pagina di sistema.



La bobina di refresh display LCD permette quando abilitata di riscrivere e aggiornare tutte le variabili che compongono una pagina.

Grazie a questa bobina si può vedere ad esempio che il conteggio di un timer aumenta. Ad esempio consideriamo lo schema seguente :



Ogni decimo di secondo il contatto speciale S.0 si chiude e con in serie il rilevatore di fronte F.1 una sola volta ogni decimo viene riscritto lo stato delle variabili della pagina attuale di visualizzazione.

Non bisogna impostare per ogni pagina nel programma un blocco di refresh perché il sistema operativo tiene traccia dell'ultima pagina visualizzata e quindi questa sarà aggiornata. Un solo blocco di refresh per programma è sufficiente a eseguire tutti gli aggiornamenti.

Il blocco refresh funziona sia sulle pagine LCD utente che LCD di sistema

Approfondimento:

La bobina refresh esegue una chiamata alla SYS Refresh pag offset 16.

Assume che la memoria R(206) sia un puntatore valido alla pagina attualmente visualizzata. Se questa memoria è impostata a zero la funzione ritorna immediatamente.



La bobina modifica pagina LCD esegue le impostazioni delle variabili presenti in una pagina.

Non bisogna impostare una bobina modifica per ogni pagina LCD presente. Il sistema operativo tiene traccia della pagina attualmente visualizzata e un solo blocco modifica è più che sufficiente per ogni programma.

Impostando le proprietà di questa bobina compare il seguente dialog box:

il campo Memoria usata deve contenere un indirizzo 300..900 per un registro usato solo da questa funzione.

Il campo divisore ammette un numero da 0 a 255 ed si riferisce per quante scansioni di programma deve essere attivo un tasto prima di effettuare una modifica.

Questa bobina usa 4 tasti posti alla destra del display e precisamente:

PIU,MENO,ESC,ENTER. Premendo ESC con PIU o MENO si ottengono variazioni più accentuate, dipende dal formato della variabile.

Quando viene visualizzata una pagina con la bobina LCD, utente o di sistema, il cursore è nascosto ma appena si preme un tasto che riguarda la modifica compare sulla variabile interessata.

Se nello stesso programma viene abilitato un blocco modifica e un blocco refresh il cursore salta, quando si ha riscrittura per l'aggiornamento delle variabili, da una variabile all'altra.

Approfondimento:

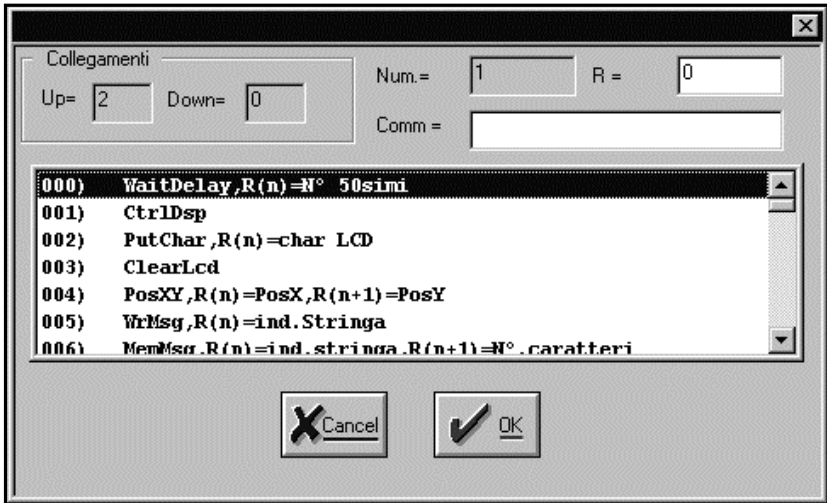
Questa bobina fa riferimento alla funzione di SYS UserInputVal offset 14.

Occorre che il registro R(206) contenga un puntatore valido ad una pagina. Se questo registro è zero la funzione ritorna immediatamente.



Le bobine SYS servono a eseguire delle chiamate alle funzioni del sistema operativo.

Impostando le proprietà di queste bobine compare il seguente dialog box:



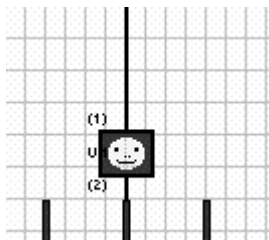
Selezionare nella lista la funzione che più interessa.

Poiché ogni funzione permette il passaggio di parametri nel campo R = impostare l'indirizzo 0...1000 del primo registro che contiene questi parametri.

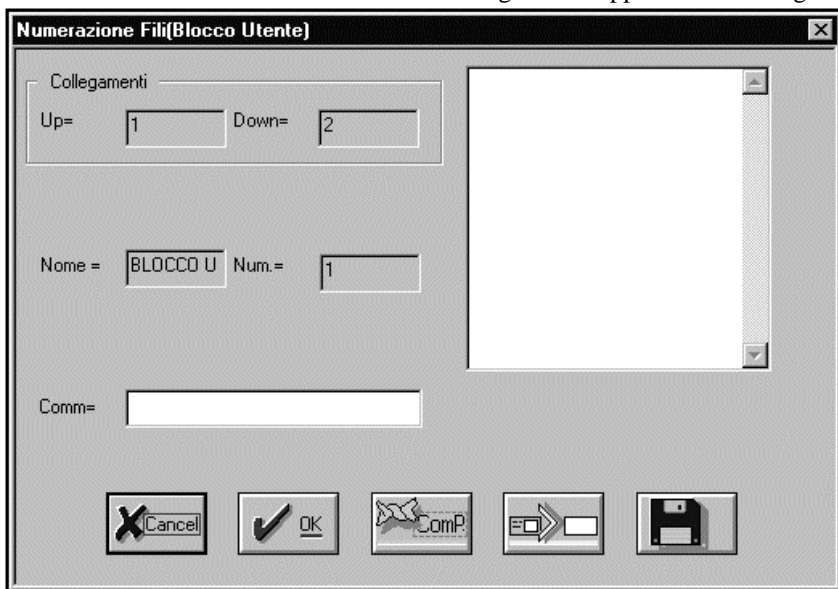
Approfondimento:

la bobina SYS esegue l'istruzione SYS #n, e quindi dipende dal tipo di funzione quanti parametri chiede e quanti ne ritorna. Nel caso di funzioni complesse è più conveniente l'utilizzo del blocco user mentre per funzioni semplici risulta più chiaro l'uso delle bobine SYS.

Il blocco user , che all'apparenza è una semplice bobina come quelle viste in precedenza, in realtà è un vero e proprio assembleatore. All'interno di questo blocco è possibile inserire tutte le istruzioni supportate dal linguaggio assembler per il microPLC merlino, e di conseguenza si aprono nuove possibilità per la risoluzione di altri tipi di problemi.. Tutte le istruzioni supportate da questo linguaggio sono elencate nel manuale d'uso Il simbolo che contraddistingue il blocco user è raffigurato nel disegno di sinistra. In un programma non ci sono limiti al numero di blocchi utilizzati se non quelli imposti dalla capacità di memoria.



Cliccando sul blocco impostato si aprirà la finestra di dialog che permette l'inserimento della parte di programma. Esaminando il disegno che rappresenta il dialog box



COMP compila la parte di programma editata all'interno della apposita finestra.

- > questo simbolo (in basso a destra di *COMP*) serve ad allargare a tutto schermo la finestra per l'inserimento delle istruzioni. Per ritornare alla dimensione originale è sufficiente ripremere nuovamente lo stesso tasto.
- Simbolo "dischetto" salva su un file specificato la parte di programma editata. Questa funzione è utile quando si vogliono conservare una serie di istruzioni che possono essere adattate a più programmi (questa procedura in gergo viene definita "MACRO").

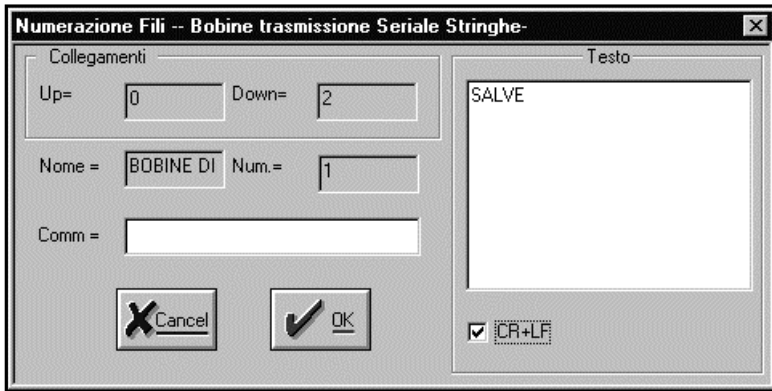
Non vi sono limitazioni all'uso del blocco user, è però necessario che le etichette usate nei singoli blocchi user siano diverse per ogni blocco.

Questo perché il compilatore non esegue nessuna modifica sul sorgente assembler presente entro un blocco user , è l'assemblatore poi a tradurlo in linguaggio eseguibile dal PLC. Per questo se poi nel sorgente assembler si trovano due etichette identiche a due indirizzi diversi viene generato un errore. Attenzione quindi con cut e copy di uno stesso blocco user che contiene etichette.

In blocchi user sono bobine a tutti gli effetti possono essere messe in parallelo.



Il blocco trasmissione testo è una bobina che quando è abilitata trasmette il testo ascii con cui è impostata. Le proprietà della bobina possono essere impostate con il sequenet dialog box:



Nel campo testo si deve inserire il messaggio da trasmettere
CR+LF se selezionato aggiunge al testo trasmesso i caratteri 13 e 10

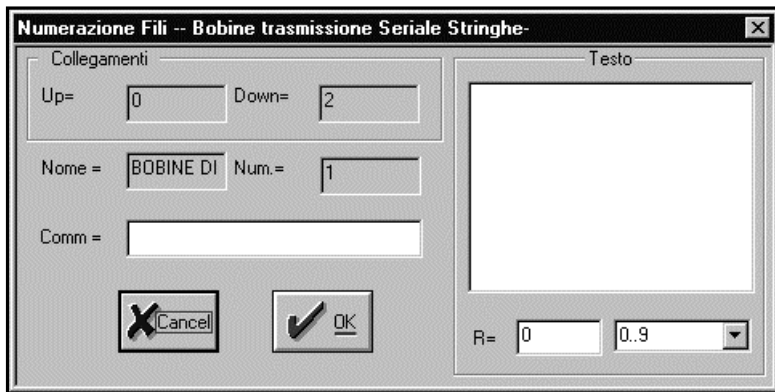
Se si abilita la bobina di trasmissione in modo stabile senza un rilevatore di fronte viene spedito un messaggio ad ogni scansione di programma. Il messaggio viene spedito tramite interrupt ma se la trasmissione è continua dopo la prima scansione il sistema operativo aspetta che il messaggio precedente venga trasmesso completamente prima di impostare il nuovo messaggio. Questo rallenta la scansione del programma.

Approfondimento:

la trasmissione avviene sulla COM 1 , e il messaggio per essere trasmesso viene copiato nel buffer di trasmissione seriale che parte da R(161). Il registro R(208) e R(211) sono impostati in modo da effettuare la trasmissione in ascii.
La ricezione binaria non viene influenzata da questa bobina.



Il blocco trasmissione testo formattato è una bobina che quando è abilitata trasmette il testo ascii con cui è impostata. Le proprietà della bobina possono essere impostate con il seguenet dialog box:



Nel campo testo si deve inserire il messaggio da trasmettere
 Nel campo R= bisogna inserire l'indirizzo 0...1000 della memoria da trasmettere.
 Nel listbox a fianco si può scegliere il formato di rappresentazione della variabile.
 Automaticamente viene aggiunto a fine messaggio il carattere 13 e 10. (CR+LF)

Se si abilita la bobina di trasmissione in modo stabile senza un rilevatore di fronte viene spedito un messaggio ad ogni scansione di programma. Il messaggio viene spedito tramite interrupt ma se la trasmissione è continua dopo la prima scansione il sistema operativo aspetta che il messaggio precedente venga trasmesso completamente prima di impostare il nuovo messaggio. Questo rallenta la scansione del programma.

Approfondimento:

la trasmissione avviene sulla COM 1 , e il messaggio per essere trasmesso viene copiato nel buffer di trasmissione seriale che parte da R(161). Il registro R(208) e R(211) sono impostati in modo da effettuare la trasmissione in ascii.
 La ricezione binaria non viene influenzata da questa bobina.



La bobina di trasmissione testo con indice trasmette un testo ascii da un array di stringhe che l'utente definisce.

Poichè la trasmissione di stringhe prende diverso spazio in memoria nel caso di molti blocchi di trasmissione seriale ascii conviene usare questo tipo di bobina che genera codice assembler più ottimizzato.

Le proprietà della bobina possono essere impostate con il seguenet dialog box:

Nei campi 1..8 si inserisce le stringhe da trasmettere .

Selezionare CR+LF se si vuole aggiungere ad ogni stringa i caratteri 13 e 10

Nel campo registro indice selezionare l'indirizzo 0...1000 del registro il cui contenuto(0...7) identifica il testo da trasmettere

Se si abilita la bobina di trasmissione in modo stabile senza un rilevatore di fronte viene spedito un messaggio ad ogni scansione di programma. Il messaggio viene spedito tramite interrupt ma se la trasmissione è continua dopo la prima scansione il sistema operativo aspetta che il messaggio precedente venga trasmesso completamente prima di impostare il nuovo messaggio. Questo rallenta la scansione del programma.

Approfondimento:

la trasmissione avviene sulla COM 1 , e il messaggio per essere trasmesso viene copiato nel buffer di trasmissione seriale che parte da R(161). Il registro R(208) e R(211) sono impostati in modo da effettuare la trasmissione in ascii.

La ricezione binaria non viene influenzata da questa bobina.



La bobina di trasmissione testo formattato con indice trasmette un testo ascii da un array di stringhe che l'utente definisce.

Poichè la trasmissione di stringhe prende diverso spazio in memoria nel caso di molti blocchi di trasmissione seriale ascii conviene usare questo tipo di bobina che genera codice assembler più ottimizzato.

Le proprietà della bobina possono essere impostate con il seguente dialog box:

Per ognuno degli 8 campi 1...8 inserire:

- Il testo da trasmettere
- L'indirizzo della variabile 0...1000 da formattare
- Selezionare il formato di visualizzazione della variabile.

Nel campo registro indice selezionare l'indirizzo 0...1000 del registro il cui contenuto identifica il messaggio da trasmettere

Se si abilita la bobina di trasmissione in modo stabile senza un rilevatore di fronte viene spedito un messaggio ad ogni scansione di programma. Il messaggio viene spedito tramite interrupt ma se la trasmissione è continua dopo la prima scansione il sistema operativo aspetta che il messaggio precedente venga trasmesso completamente prima di impostare il nuovo messaggio. Questo rallenta la scansione del programma.

Approfondimento:

la trasmissione avviene sulla COM 1, e il messaggio per essere trasmesso viene copiato nel buffer di trasmissione seriale che parte da R(161). Il registro R(208) e R(211) sono impostati in modo da effettuare la trasmissione in ascii.

La ricezione binaria non viene influenzata da questa bobina.



Il blocco di ricezione seriale ascii è una bobina identifica una stringa ricevuta via seriale.

Per implementare questa operazione viene rimappato l'interrupt della COM1. Solo una bobina di ricezione seriale può essere usata per ogni programma . Selezionando le proprietà appare il seguente dialog box:

Questa bobina è abbastanza complessa e richiede:

2 registri riservati per la gestione dell'interrupt seriale, impostare l'indirizzo del primo di questi registri nel campo appropriato.

1 registro di uscita che viene impostato a 1 appena la stringa viene ricevuta.

Nel campo in alto selezionare la stringa da decodificare.

Approfondimento:

Questa bobina usa la SYS SetUserSvrCom1 offset 42 e quindi la comunicazione binaria di default con il personal computer viene esclusa. La memoria R(297) indica lo spiazzamento sul programma utente della routine di interrupt. Dopo aver piazzato una bobina di ricezione seriale che va sempre in testa al programma bisogna reimpostare l'inizio della scansione del programma tramite una bobina go.



La bobina di ricezione stringhe ascii da un array di stringhe permette di monitorare la ricezione di diversi messaggi contemporaneamente.

Per implementare questa operazione viene rimappato l'interrupt della COM1. Solo una bobina di ricezione seriale può essere usata per ogni programma . Selezionando le proprietà appare il seguente dialog box:

Numerazione Fili -- Bobine Ricezione Seriale Stringhe

Collegamenti

Up= 1 Down= 2

Nome = BOBINE DI Num. = 1

Comm =

1)

2)

3)

4)

5)

6)

7)

8)

Reg. riservati interrupt = 0 Reg. Out. 0

Cancel OK

Nei campi 1..8 impostare le stringhe da ricevere.

Impostare quindi l'indirizzo dei registri riservati per la gestione dell'interrupt (2). E il registro di uscita (bit 0...7).

Approfondimento:

Questa bobina usa la SYS SetUserSvrCom1 offset 42 e quindi la comunicazione binaria di default con il personal computer viene esclusa. La memoria R(297) indica lo spiazamento sul programma utente della routine di interrupt. Dopo aver piazzato una bobina di ricezione seriale che va sempre in testa al programma bisogna reimpostare l'inizio della scansione del programma tramite una bobina go.



Il contatto variazione di un registro è un contatto che da un impulso di una scansione di programma ogni volta che la parte bassa (Low Word) del registro assegnato cambia.

Questo tipo di contatto copia la parte bassa del registro nella parte alta del registro quando sono diverse e genera l'impulso.

Le proprietà di questo contatto sono impostabili tramite il seguente dialog box:

Numerazione Fili -- Contatti su Variazione Registro-

Collegamenti

Up= Down=

R=

Nome = Num. =

Comm =

questa funzione lavora sui primi 16 bit

Nel campo R= specificare l'indirizzo 0...1000 del registro interessato.



La bobina di salto è una bobina che quando abilitata interrompe la normale scansione di programma , facendola passare alla etichetta su cui è impostata la bobina.

Le proprietà sono impostabili tramite il seguente dialog box:

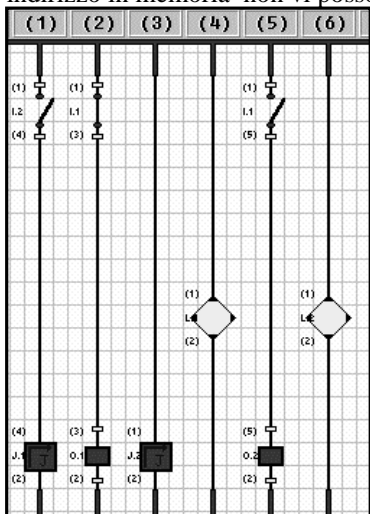


Nel campo Salto All'Etichetta inserire il numero di etichetta a cui fa riferimento la bobina.



Quella rappresentata a fianco è la bobina etichetta .

Ogni volta che si piazza nel disegno uno di questi componenti automaticamente viene numerato in modo crescente. Poiché ogni etichetta è un indirizzo in memoria non vi possono essere nello schema due etichette con lo stesso numero.



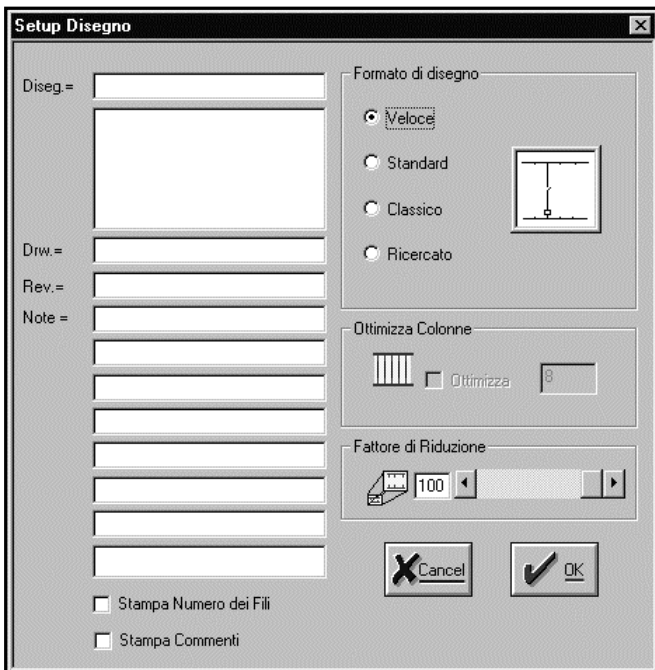
Analizziamo il seguente schema:

Alla partenza del programma se non si attiva nessuno ingresso la colonna 1 non abilita il salto della bobina J.1 impostata sull' etichetta L.1.La colonna 2 attiva l'uscita O.1 .La bobina di salto della colonna 3 è sempre attiva e esegue un salto di programma alla colonna 6 perche la bobina J.2 è impostata sulla etichetta L.2. in questo stato le colonne 4 e 5 non sono neanche elaborate. In questo stato O.1 rappresenta lo stato Invertito di I.1.

Attivando I.2 la bobina J.1 è attiva .Adesso vengono ignorate le colonne 2 e 3 e il programma elabora la sola colonna 5 che ha come risultato la copia di I.1 in O.1

Tramite il programma è possibile stampare il disegno ottenuto .Selezionare dal menu

File Stampa.Compare il Seguento Dialog Box:



Sono Possibili 4 formati di stampa:
Veloce:

Vengono disegnati solo i componenti in modo continuo e con il trattore sulla stampante si ottiene un disegno non interrotto.

Standard:

Il disegno viene diviso in pagine,dove sono riportati il numero delle colonne e delle righe:

Classico:

Il disegno viene scalato nel formato in modo da visualizzare sia i dati

anagrafici della ditta che ha composto il disegno che le revisioni:

Ricercato:

Il disegno viene scalato nel formato in modo da visualizzare sia i dati anagrafici della ditta che ha composto il disegno che le revisioni,che le note .

Il campo Ottimizza colonne se impostato con Ottimizza elimina le parti di disegno vuote rappresentando ingrandito il disegno.

Il fattore di riduzione scala in percentuale il disegno stesso.

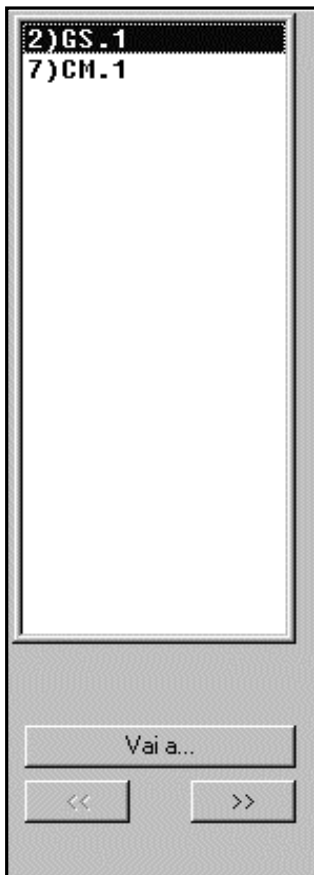
Stampa numero di fili se abilitato disegna anche tutti i numeri di fili.

Stampa commenti inserisce nel reference dei componenti anche i commenti impostati dall'autore.

I campi di destra sono campi ascii che possono essere impostati e vengono riportati nel disegno a secondo della stampa impostata.

La leggibilità di una stampa dipende dai componenti utilizzati , se si è usato ingressi uscite o componebtì generici di base è chiaro che la stampa rispetta esattamente il programma ma se ad esempio si è usato blocchi LCD non viene visualizzato nella stampa la struttura costituente la pagina.

Per questo è utile corredare la stampa di un programma su schema elettrico anche con la stampa del sorgente assembler ottenuto.



Selezionando dal menu Edit Vai a .. compare il toolBox per spostarsi velocemente sul disegno.

Compare l'elenco completo di tutte le bobine presenti nel disegno.

Selezionare con il mouse la bobina interessata e inviare il tasto Vai a....

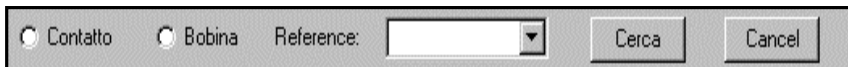
Automaticamente il disegno viene spostato alla colonna della bobina selezionata.

Per chiudere Questo toolBox o in genere qualsiasi toolBox del programma portarsi sopra il toolBox selezionato con il mouse e inviare il tasto destro del mouse.

Compare un Menu PopUp con tutti i ToolBox presenti e quelli visibili selezionati.

Scegliere tra i toolbox selezionati quello da far scomparire e tra quelli non selezionati quello da visualizzare.

Se da menu Edit si seleziona cerca componente compare il seguente toolBox:



Se si vuole ad esempio cercare tutti i contatti del Rele R.1 selezionare il campo Contatto e nel campo Reference impostare R1. Inviare il bottone Cerca. Compare un nuovo ToolBox con tutti i riferimenti ad R.1:



Inviando il tasto vai (tasto con il cannocchiale) il disegno

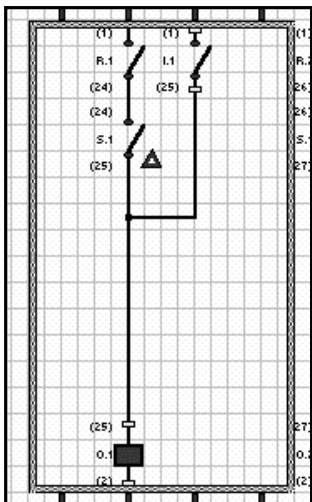
viene spostato alla colonna e riga (Nell'esempio Col.129, Riga B). Premere i tasti Cancel o quello a fianco di Vai per far scomparire il toolbox



Tramite il comando Esporta macro si può salvare una parte di disegno per poi poterla reimportare .

Procedere nel seguente modo.

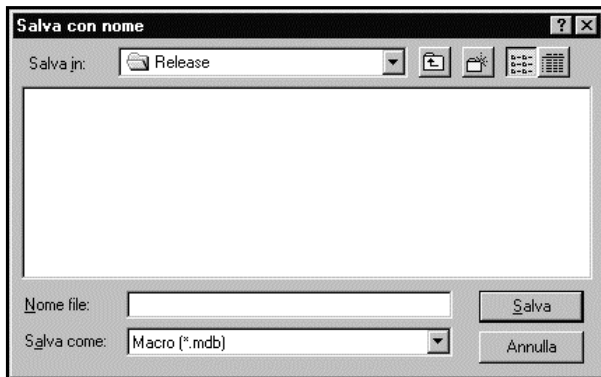
Selezionare con il mouse l'area da esportare premendo il tasto sinistro e spostandosi. I contorni che delimitano la selezione variano come in figura:



A questo punto il tasto esporta macro da disabilitato si abilita.

Inviare il comando Esporta macro.

Compare il dialog box per la gestione dei file impostato sulla estensione macro:



Dare un nome valido al blocco da esportare e inviare Salva.

Tramite il comando importa macro si può inserire una parte di programma in precedenza salvata con il comando esporta macro.



Inviare il comando ,non succede nulla.

Cliccare sopra la colonna dove iniziare l'inserimento.



Compare il dialog box per la gestione dei file già impostato per la apertura di un blocco macro.

Selezionare il file desiderato e inviare apri

Tutto il disegno contenuto nella macro viene riversato nel disegno corrente.

Attenzione l'area deve essere vuota altrimenti il disegno importato può sovrascrivere il disegno corrente.

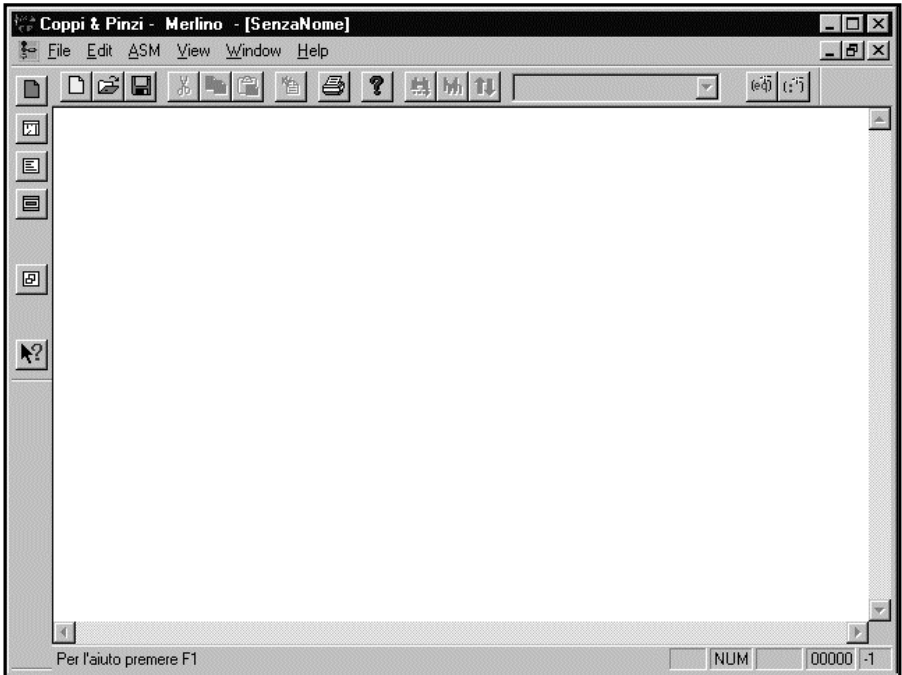
Tutti i componenti importati dalla macro vanno reimpostati sui componenti il disegno. Questo perché le macro devono essere disegni di concetto, supponiamo di avere come macro il disegno di un timer ciclico, quindi viene importato lo schema corretto di un timer ciclico ma tutti i riferimenti sono da impostare perché su un programma si può usare T.1 e T.2 mentre su un altro T.8 e T.23 per un timer ciclico.

Ogni disegno compilato come abbiamo visto genera un sorgente assembler che poi a sua volta genera il codice eseguibile per il PLC.



Si può altresì generare direttamente un sorgente assembler inviando il comando Nuovo Assemblatore

Compare un nuovo tipo di finestra,assemblatore appunto con propri toolbar e una pagina bianca vuota.



dove i bottoni del toolbar sono associati ai seguenti comandi



Nuovo File



Apri un file



Salva il file corrente



Assembla il file corrente



Cut .Copy ,Paste



**Find.Maiuscole minuscole,in Su
inGiu**

**Toolbox definizioni****ToolBox etichette**

Si può editare direttamente istruzione dopo istruzione il programma da comporre. Il programma si comporta per questa fase come un qualsiasi text editor.

Per facilitare alcune funzioni sono disponibili diversi toolbox.

Uso comune in programmi complessi è associare a un numero una definizione ad esempio

ACC equ R(300)

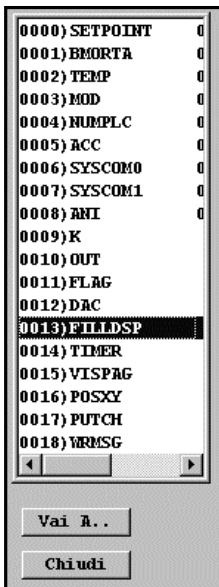
Associa al testo ACC il testo R(300) e quindi si può scrivere un passo di programma così composto:

```
MOV.L #0xfe89ace,ACC
```

E per il programma assemblatore sarà come avere

```
MOV.L #0xfe89ace,R(300).
```

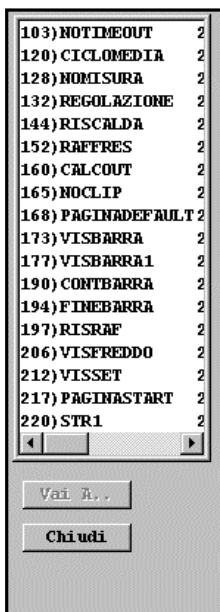
Quando vi sono molte definizioni si può premere sul toolbox definizioni e compare il seguente toolbar:



Vengono visualizzate tutte le definizioni presenti nel programma corrente. Selezionandone una qualsiasi e inviando il comando vai a il cursore nella pagina testo si sposta direttamente sulla riga dove è la dichiarazione della definizione. Inviare il comando Chiudi per far scomparire questo toolbar. In modo analogo è presente il toolbox della lista di tutte le etichette presenti nel programma corrente.

Inviare dalla barra comandi toolbox etichette.

Compare il seguente toolbar:



Analogamente al toolbar delle definizioni selezionando un item dalla lista con il comando vai al cursore nella pagina del testo si posiziona alla linea cercata.

Con il tasto chiudi scompare il toolbar.

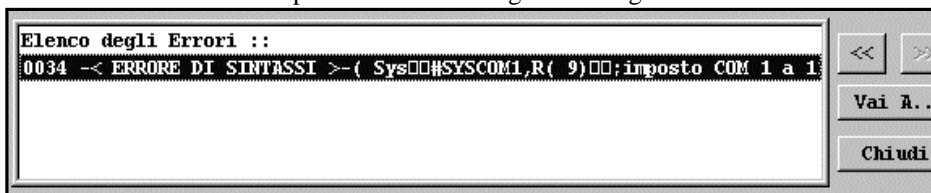
Una volta costruito il proprio programma si assembla con l'apposito comando da toolbar generale.

Se non vi sono errori presenti compare il seguente dialog box:



Selezionando il campo Simulatore la finestra corrente viene minimizzata e viene automaticamente aperto una sessione simulatore che permette di caricare sul plc il codice eseguibile del programma assemblato.

Se invece vi sono errori compare il toolbar della gestione degli errori.



Selezionando un item dall'elenco degli errori i tasti vai a successivo e precedente si abilitano. Inviando uno di questi comandi il cursore nella pagina testo si sposta direttamente alla linea dove è stato generato l'errore. Selezionare Chiudi per far scomparire questo dialog.box.

Il comando find funziona abbinato o alla parola selezionata nell'area di digitazione del testo o immettendo direttamente nel list box del toolbar la chiave di ricerca.

Nel primo caso selezionare premendo il tasto shift e muovendosi con le frecce la parola da ricercare e inviare il comando find. La parola selezionata viene copiata nel listbox delle parole ricercate e inizia la ricerca.

Nel secondo caso dovo avere digitato nel listbox delle parole ricercate la nuova chiave inviare il comando find e inizia la ricerca.

Il tasto Minuscolo maiuscolo quando non è premuto (il tasto è stabile e se si preme rimane premuto) permette la ricerca della parola senza il controllo maiuscolo minuscolo altrimenti attiva questo filtro.

Il tasto ricerca in giu o in su se non è premuto permette la ricerca solo in giu altrimenti solo in su.

Nel menu edit sono disponibili anche i dialog box cerca, cambia e ripeti. La implementazione di questi menu segue fedelmente quella standard di window per cui non vengono in questa sede spiegati.

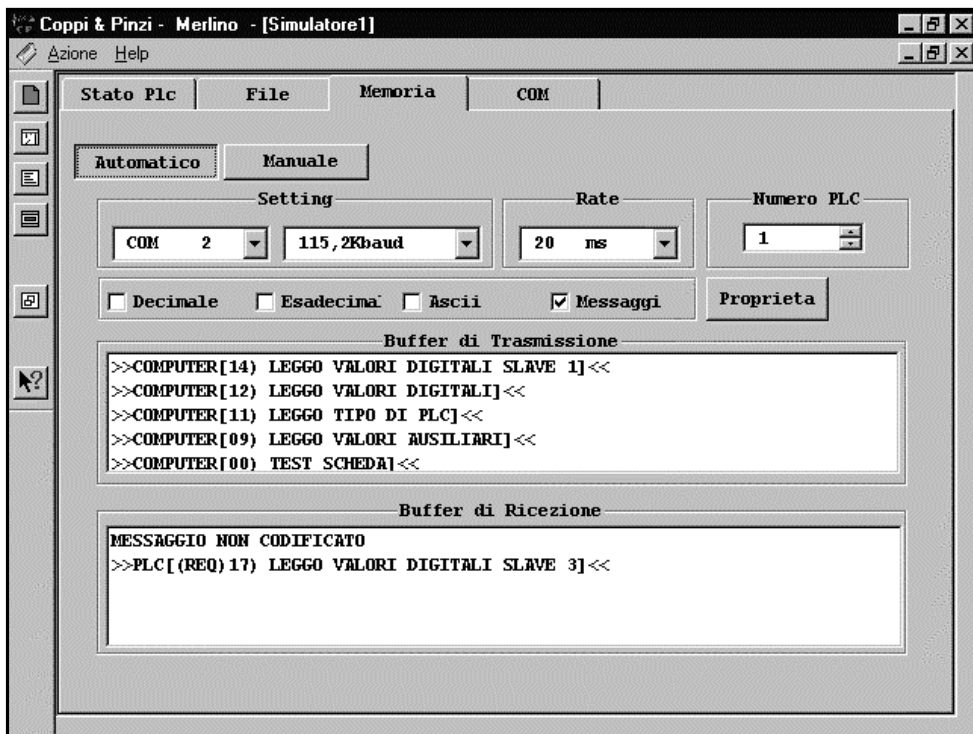
Il campo proprieta del disegno permette di vedere la data di creazione del programma nonché la lunghezza stessa del file.

A questo proposito bisogna dire che non si può editare programmi più lunghi di 64k . Considerando le capacità di memoria del merlino questa non è una limitazione.

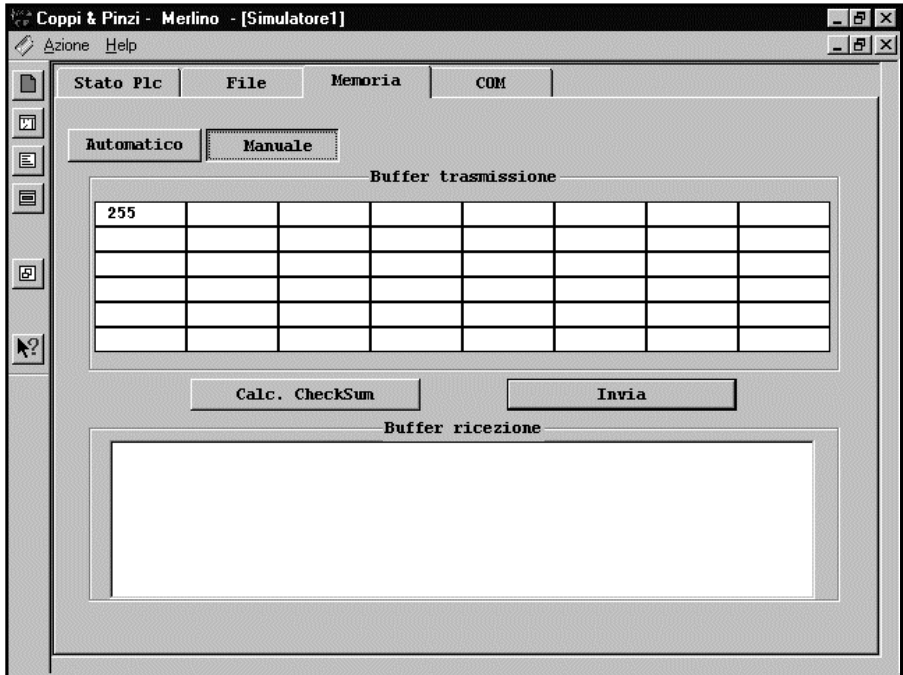
Nel menu View sono presenti i campi Screen font e printer font.

Il primo permette di impostare il font di visualizzazione video l'altro il font di stampa. E' buona norma impostare un font a spaziatura fissa che rende più leggibili i programmi ad esempio courier new o system.

Ogni volta che si apre una nuova sessione di simulazione compare la seguente finestra



Il programma in modo automatico cerca la prima porta di comunicazione seriale disponibile e la apre. Se la porta non dovesse essere quella interessata dalla comunicazione con il PLC selezionare nel campo setting la porta desiderata e la velocità richiesta. Ricordiamo che ogni volta che il PLC viene forzato in stop prende la velocità di 115.200 baud. Il campo rate informa il sistema ogni quanto tempo in millisecondi processare i messaggi ricevuti dal PLC. Mentre nel campo Numero di PLC va impostato il numero del PLC da interrogare nella rete. I campi decimale, esadecimale, messaggi e ascii impostano il modo di visualizzazione con cui rappresentare negli editbox sottostanti i messaggi trasmessi e ricevuti dal PLC. Premendo il tasto manuale compare una nuova finestra che permette di formattare manualmente i messaggi da spedire direttamente a un PLC della rete e analizzare il messaggio di ricezione:



Negli edit disponibili nel campo buffer di trasmissione si può impostare i numeri costituenti il messaggio da formattare. Se il messaggio si attiene al protocollo di comunicazione del PLC allora inviando il tasto Calc.CheckSum automaticamente viene calcolata la corretta checksum per il messaggio editato.

Premendo il tasto invia si spedisce il messaggio composto e si controlla il buffer di ricezione. Quando si riceve un messaggio il programma lo visualizza in formato numerico nell'editbox buffer di ricezione.

Se nel selettore delle cartelle si seleziona il campo File



Compare la finestra per la gestione e programmazione della memoria flash del PLC.

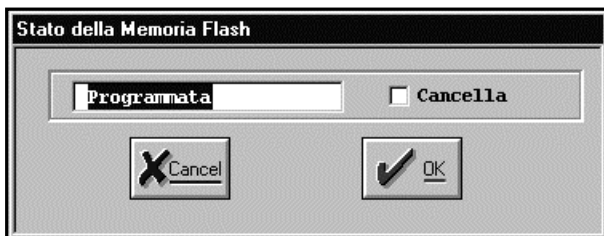


Le barre posizione di lettura e posizione di scrittura sono abilitate sono durante una scrittura o lettura della memoria e indicano l'indirizzo corrente sotto

processo.

I 4 tasti in alto possono essere abilitati o disabilitati dipende dallo stato della memoria flash. Se la memoria è vuota il tasto scrivi è abilitato e cancella disabilitato altrimenti è l'opposto.

In questo caso premendo il tasto cancella compare il dialog box per confermare questa azione :



Abilitare il campo cancella e inviare il tasto OK.

La memoria sul plc viene cancellata ,Il PLC esegue un test e si riavvia .Il programma in memoria del PLC va irrimediabilmente perso.

Premendo invece il tasto Leggi compare il seguente dialog:

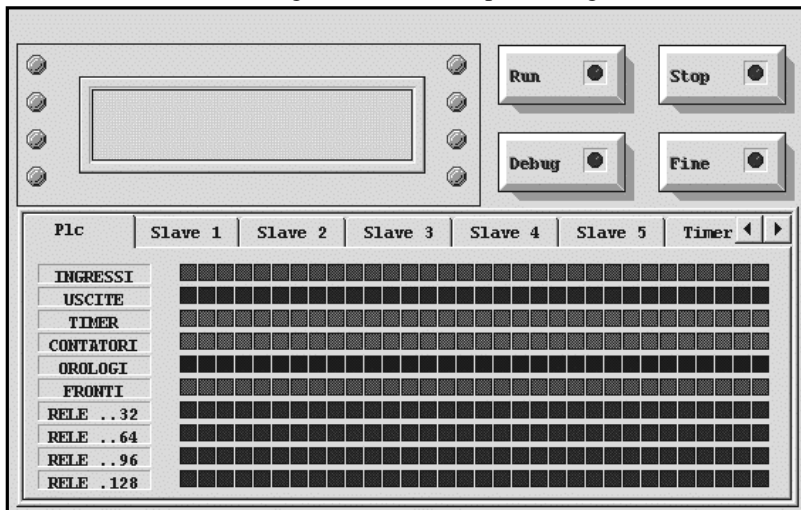


Selezionare il range corretto della memoria da leggere ma se non si è sicuri è meglio lasciare gli indici di figura con cui l'intera memoria programma viene letta. Dopo che è stata effettuata una lettura il programma residente nel PLC è in memoria del Personal Computer. Per salvarlo in un file da menu Azione inviare il comando Salva come.

Compare il dialog box per la gestione dei file, dare un nome corretto con estensione epc (eseguibile) e inviare il comando salva. Il tasto con il punto interrogativo serve per vedere se il programma in memoria del PLC corrisponde con quello residente in memoria del personal computer.

Nel caso questa condizione non si verificasse compare il messaggio Interrotto per errori.

Se dalla selezione cartelle si sceglie Stato Plc compare la seguente finestra:



Premendo il tasto RUN il PLC esegue il programma corrente in memoria.

Premendo il tasto stop la esecuzione del programma sul PLC termina.

Premendo il tasto fine viene terminata la sessione attuale di simulazione.

Il display virtuale è attivo e rispecchia per quanto possibile lo stato reale del display sul PLC.

In basso sono presenti molte cartelle per monitorare lo stato di diversi componenti.

Le prime 5 però ,PLC Slave 1...5 , permettono anche di impostare lo stato di una qualsiasi bobina rappresentata .Basta ad esempio cliccare su un qualsiasi quadrato presente (es R.1) complementare lo stato attuale della bobina.

Premendo il tasto debug il PLC passa dal modo RUN a DEBUG REMOTO.

La parte bassa della finestra si modifica nel seguente modo:

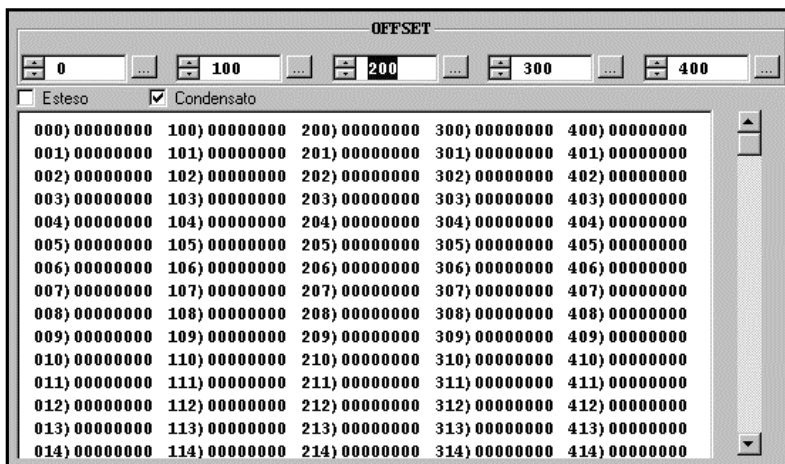


Attraverso questa finestra si può eseguire una istruzione per volta, anal

izzare lo stato dei registri sorgente o destinazione impostare i breakpoint ed andare in RUN simulato passo passo.Premendo nuovamente RUN o Stop si ritorna il condizioni normali.

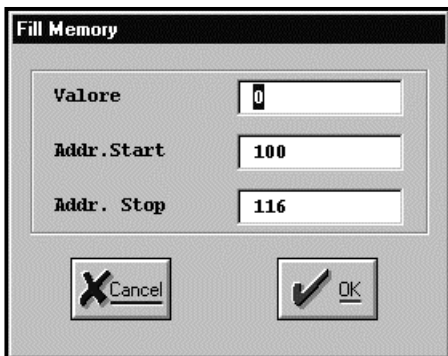
Selezionando invece la cartella memoria compare la parte più interessante per lo sviluppo di programmi ovvero il data base interno del PLC.

Viene visualizzata la seguente finestra:



In

formato condensato sono visibili 5 colonne da 16 memorie che vengono continuamente aggiornate .Oggni volta che lo stato di una memoria cambia passa da colore blu a rosso.Nei campi offset si può impostare l'indirizzo di partenza per ogni colonna.Spostando lo scrollbar laterale tutti gli indirizzi vengono aumentati o diminuiti .I tasti rappresentati con i puntini permettono di impostare la memoria ad un determinato valore.Premendo uno qualsiasi di questi tasti compare il dialog box:



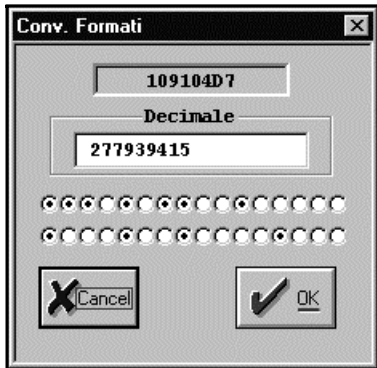
Impostare i vari campi a proprio piacimento e dare OK.La memoria nel range specificato viene impostata. Ogni valore rappresentato della memoria può essere modificato cliccando con il mouse sul valore da modificare .premere cancella e digitare il nuovo valore. Quindi premere invio per abilitare la modifica .

Per facilitare la stesura di numeri si ha un convertitore di formato.

Eseguire un doppio click con il mouse su

una qualsiasi memoria visibile.

Compare il dialog box:



I vari campi sono attivi e i cerchietti si riferiscono ai 32 bit costituenti una singola memoria. Premendo OK alla memoria corrispondente viene Assegnato il valore presente nel convertitore di formati. Per effettuare la modifica premere invio.

Selezionando il campo Esteso la finestra di visualizzazione del data base fa vedere le prime 300 memorie ,quelle di sistema , con a fianco la specializzazione di ciascuna. Es:

000)	SYS_ADDRSTART	(PROGR. MEM.FLASH)	00000000
001)	SYS_ADDRSTOP	(PROGR. MEM.FLASH)	00000000
002)	SYS_FLAG	(PROGR. MEM.FLASH)	00000000
003)	SYS_NUMPLC	(NUMERO DI PLC LATO HOST)	00000000
004)	SYS_ERRPLC	(ERRORI PRESENTI NEL PLC)	00000000
005)	SYS_STATOPLC	(STATO DEL PLC RUN. .STOP)	00000000
006)	SYS_FLAGOUT	(USCITE ABIL. /DISABIL.)	00000000
007)	SYS_VELCOM	(VELOCITA COM.SERIALE)	00000000
008)	SYS_CHIAVE	(PASSWORD)	00000000
009)	SYS_ACC	(ACCUMULATORE PER OP.BIT)	00000000
010)	SYS_PC	(CONTATORE DI PROGRAMMA)	00000000
011)	SYS_CS	(CONTATORE DI STACK)	00000000
012)	SYS_FUP	(MEM. PER RILEV. FRONTI)	00000000
013)	SYS_START	(IND. START PRORAMMA)	00000000
014)	SYS_DEBUG 0	(MEM. APPOGGIO PARAMETRI)	00000000

**AGGIORNAMENTO
FINO ALLA
VERSIONE
2.76**

OTTOBRE 1999

Numerose modifiche sono state apportate all'ambiente di programmazione, sempre comunque mantenendo piena compatibilità con le versioni precedenti.

In particolare sono state effettuate tutte quelle modifiche atte a migliorare l'interfaccia, ad esempio impostando i ToolBar sullo style windows98, riprogettando la stampa, impostando la trasmissione di documenti etc.

La modifica più importante è senz'altro che il programma serve come ambiente di programmazione per il microPLC merlino e il microPLC lilith.

Quindi la prima azione da effettuare è quella di selezionare il tipo di PLC con cui si sta lavorando.

Per effettuare questa selezione sono stati inseriti due nuovi bottoni nella barra generale usualmente posta a sinistra. I due bottoni inferiori selezionano:

1. **MicroPLC Merlino**
2. **MicroPLC Lilith**

Tali bottoni sono a stato fisso per cui la selezione corrente imposta lo stato del bottone come premuto. Ogni volta che si esce dal programma la selezione corrente del PLC viene memorizzata nel file di inizializzazione in modo che alla prossima apertura l'ambiente sia già pronto. In corrispondenza di ogni selezione lo sfondo della finestra principale visualizza l'immagine del microPLC selezionato.



Non sono state effettuate ulteriori modifiche nella finestra generale del programma

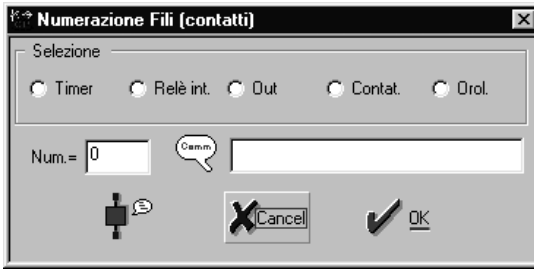
• NUOVI DIALOGBOX

Sono stati riprogettati completamente i dialog box del programma in modo da ottenere dialog più sintetici ed efficienti. In particolare ogni dialogBox del disegno ha al suo interno un bottone in stile Flat che quando premuto richiama un dialog con i collegamenti del blocco. In questo modo ogni dialogBox è stato eliminato i campi per i collegamenti il nome e dove possibile per il numero.

Ad esempio

Il dialog box dei contatti generici è passato da :

Al seguente dialog



- **NUOVI BLOCCHI DI PROGRAMMAZIONE**

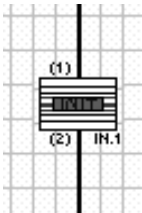
Sono stati inseriti nuovi blocchi di programmazione che vengono di seguito spiegati ad uno a uno. Questi Blocchi sono:

1. Blocco INIT
2. LCD4X20
3. MENU
4. BOBINE GENERICHE
5. BOBINE GENERICHE MONOSTABILI
6. BOBINEGENERICHE SET
7. BOBINE GENERICHE RESET
8. BOBINE RANGE ANALOGICO
9. BOBINE MEDIA ANALOGICA
10. CONTATTI TASTI (LILITH)

- **MIGLIORIE GENERICHE**

A discrezione della C&P sono state apportate migliorie sul codice o sul sistema. Ad esempio sono stati riprogettati i menu di contesto o il piazzamento dei componenti.

• BLOCCO INIT



Essenzialmente il blocco init serve per impostare i parametri di collegamento e la presenza di PLC slave nella rete del PLC. Sostituisce il menu Inizializzazione del campo edit in quanto questa impostazione pur essendo valida non viene registrata nel file disegno costringendo a reimpostare i parametri di inizializzazione ad ogni apertura – modifica. Una volta inserito un blocco Init nel disegno viene disabilitato il menu iniziale nel campo edit, in quanto il blocco init ha priorità maggiore.

Selezionando le proprietà del blocco compare il seguente dialogBox per le impostazioni dei parametri:

The screenshot shows a dialog box titled 'Inizializzazione'. It contains the following fields and controls:

- Numero PLC = 1 (text input)
- COM HOST: Vel Com. = 115200 (dropdown menu)
- COM SLAVE: Vel Com. = 115200 (dropdown menu)
- PassWord: 0 (text input)
- Output: Disabil Abilit.
- A table with columns 'Registro', 'Valore', and 'Com'.
- Buttons: Cancel (with an X icon) and OK (with a checkmark icon).

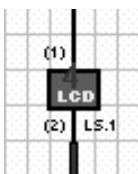
Si può impostare:

Il numero PLC, la velocità di comunicazione tra PLC e P.C (host), la velocità di comunicazione tra PLC e i propri slave (slave) la password di sistema per impostazione dei parametri e se le uscite debbano partire come abilitate allo start.

Nel campo inferiore si può impostare dei registri del PLC al valore desiderato come default. Il compilatore inserisce del codice che imposta questi registri al valore desiderato prima dello start program. (ad esempio se il PLC ha una rete di slave bisogna impostare in questo campo la impostazione dei registri presenza slave al valore desiderato).

Questo blocco può essere usato anche come blocco START migliorato. In questa funzione infatti Permette la impostazione prima dello start dei registri specificati

- **LCD 4X20**



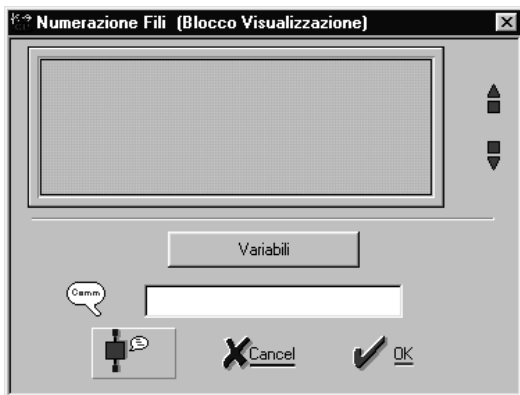
Questa bobina è specifica del microPLC Lilith e supporta pagine virtuali fino a 256 righe per 20 caratteri con variabili (quante la memoria ne supporta).

Attenzione:

questa bobina è specifica del microPLC LILITH se viene usata sul microPLC MERLINO non viene visualizzata nessuna stringa.

L'uso di questo tipo di bobina è identico a quello di un lcd standard a 2x16 caratteri. Una volta che la logica abilita questo blocco la pagina viene stampata sul display e rimane anche quando la corrispondente bobina è disabilitata (Viene memorizzata l'ultima pagina stampata).

Impostando le proprietà di pagina compare il seguente dialog Box:



Questo dialogo è identico come funzionamento a quello per LCD 2X16 caratteri solo che il display che compare è 4X20 caratteri si può quindi impostare messaggi di questa dimensione. Inoltre premendo i tasti freccia posti alla destra del LCD è possibile inserire nuove righe oltre le 4 visualizzate. La gestione di queste righe viene lasciata al sistema operativo che automaticamente esegue lo scroll del display quando si deve impostare una variabile fuori pagina.

Attenzione: lo scroll è abilitato solo se vi sono delle variabile fuori pagina se ad esempio la variabile 5 è sulla riga 5 allora quando andiamo a impostare la

variabile attraverso un blocco modifica le scritte vengono shiftate in alto quando passiamo dalla modifica della variabile 4 alla variabile 5 mentre vengono shiftate in basso quando si passa dalla modifica della variabile 5 alla variabile 1. **Per ottenere scritte che sono shiftate, senza variabili bisogna usare il blocco menu.**

Premendo il tasto variabile si passa alla impostazione delle variabili di pagina. Compare il solito dialog box del LCD 2X16:

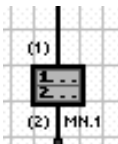


l'unica differenza è la seguente:

mentre il display LCD 2X16 il valore Y è bloccato come massimo a 1 (0,1) in questo caso dipende dal numero di righe presenti nella pagina. Ad esempio se si è inserito 6 righe allora Y può assumere come valore massimo 5 (0..5).

In ogni caso prima bisogna inserire i messaggi di testo in modo da impostare indirettamente anche il massimo delle righe ammesse poi inserire le variabili.

• BLOCCO MENU



Il blocco menu è comune per ambedue i tipi di PLC e ha lo stesso funzionamento, solo quando è impostato il microPLC Lilith il display che compare per la impostazione dei messaggi ha una ampiezza di 4 righe per 20 caratteri.

Altra differenza che però è trasparente per il programmatore è che mentre per il microPLC merlino il blocco viene implementato da programma con istruzioni appropriate, per il microPLC Lilith è stata usata una apposita funzione di sistema in modo da ottimizzare il processo.

Impostando le proprietà di questa bobina compare il seguente dialogBox:



Si può inserire i messaggi direttamente sul display, usando le frecce poste a sinistra dello stesso per shiftare in alto le scritte.

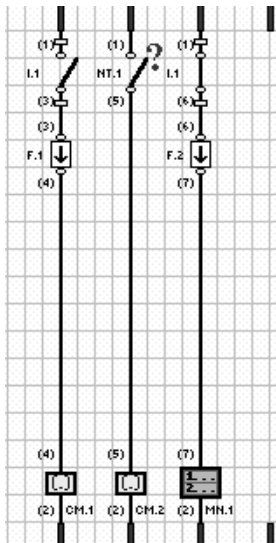
Tramite il tasto cancella viene eliminato l'ultimo messaggio inserito, mentre con cancella tutto vengono eliminati tutti i messaggi inseriti.

Con il pulsante Lista compare un dialog in cui sono visualizzati tutti i messaggi inseriti.

Impostando il campo selezione la prima colonna del display deve essere lasciata vuota per far posto alla freccia che compare in modo automatico.

Nel registro indice bisogna inserire l'offset del registro di memoria usato

come indice delle stringhe da visualizzare. Se il display è di 2X16 con il contenuto del registro indice uguale a zero vengono visualizzate le prime due righe, se il registro è 1 viene visualizzato la seconda e la terza e così via.



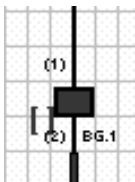
Vogliamo in questa sede ricordare il circuito minimo per l'utilizzo di un menu:

Quando si preme I1 per una volta il blocco CM1 incrementa il registro di selezione R(300) di uno ovvero $R(300)=R(300)+1$

Il test NT1 controlla che questo registro sia nel range specificato esempio 8 ovvero si chiude quando $R(300)>8$. In questo caso il blocco CM2 imposta il registro R(300) a zero (menu circolare).

Infine il contatto N.C. di I1 abilita il menu per una volta (f2) in occasione di ogni variazione di I1 in modo da visualizzare le righe corrispondenti al registro indice R(300).

- **BOBINE GENERICHE**



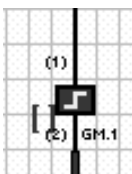
Questo tipo di bobine possono essere assegnate a qualsiasi bit di un qualsiasi registro del sistema. In questo modo si può ovviare a diversi problemi che si possono incontrare nella programmazione. Ad esempio uso di rele a ritenzione di stato oltre quelli già definiti nel sistema (129-256), formattazione di vari stati su un unico registro etc.

La bobina generica normale ha lo stesso funzionamento di un rele. Considerando che lo stato della bobina può essere letto con un contatto generico impostato sullo stesso bit e registro appare chiaro che il Merlino ha a disposizione 32.000 rele (1000 Registri long word).

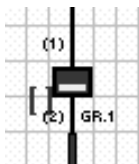
Analogamente ai rele le bobine generiche possono essere monostabili, reset o set.

Quando si impostano le proprietà di un simile componente compare il dialog box dei contatti generici che permette di impostare il numero di registro e di bit a cui è impostata la bobina.

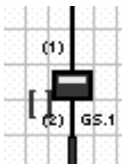
- **BOBINE GENERICHE MONOSTABILI**



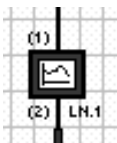
- **BOBINE GENERICHE RESET**



- **BOBINE GENERICHE SET**



• BOBINE RANGE ANALOGICO



Tramite queste bobine è possibile impostare la conversione in un range specificato un ingresso analogico qualunque.

Attenzione:

Per default i microPLC eseguono la conversione Analogico digitale alla frequenza di circa 10 conversioni a secondo quindi non ha senso inserire direttamente questa bobina tra i fili di alimentazione in quanto per molte scansioni di programma il valore convertito è sempre lo stesso. Per acquisizioni ad alta velocità occorre prima di questa bobina chiamare direttamente la funzione SYS di conversione analogico digitale. In genere è buona norma inserire un contatto speciale con in serie un fronte sulla abilitazione della bobina.

Visualizzando le proprietà della bobina compare il seguente dialogBox:

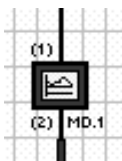
Numerazione Fili -- Scala Ingressi Analogici --

<input checked="" type="checkbox"/>	Converti An.1 da..	0	a	500	su Reg.	300	<input checked="" type="checkbox"/>	Media Val.Prec.
<input checked="" type="checkbox"/>	Converti An.2 da..	100	a	300	su Reg.	301	<input checked="" type="checkbox"/>	Media Val.Prec.
<input type="checkbox"/>	Converti An.3 da..		a		su Reg.		<input type="checkbox"/>	Media Val.Prec.
<input type="checkbox"/>	Converti An.4 da..		a		su Reg.		<input type="checkbox"/>	Media Val.Prec.
<input type="checkbox"/>	Converti An.5 da..		a		su Reg.		<input type="checkbox"/>	Media Val.Prec.
<input type="checkbox"/>	Converti An.6 da..		a		su Reg.		<input type="checkbox"/>	Media Val.Prec.
<input type="checkbox"/>	Converti An.7 da..		a		su Reg.		<input type="checkbox"/>	Media Val.Prec.
<input type="checkbox"/>	Converti An.8 da..		a		su Reg.		<input type="checkbox"/>	Media Val.Prec.

Per ogni canale analogico si può impostare il valore di Zero (1° colonna) di Span (2° colonna) l'offset per il registro di conversione e se si vuole mediare con la conversione precedente. Ad esempio nella prima riga viene convertito il canale analogico 1 in un range da 0 a 500 sul registro 300 eseguendo la media con il valore precedente.

• BOBINE MEDIA ANALOGICA

Tramite le bobine media analogica si converte un canale analogico in un range specificato ed in più si imposta una media dei valori del peso specificato.



Anche per questa bobina vale quanto detto per la bobina range analogico a proposito della abilitazione a tempo.

Impostando le proprietà della bobina compare il seguente dialogBox:



Come si vede dalla figura viene richiesto :

numero di canale analogico da convertire nel range con la media

valore dello Zero

valore dello Span

il registro di USCITA (r(300))

due registri : il primo come contatore dei valori mediati, il secondo come somma dei valori

Il numero di valori sommati prima di eseguire la divisione per la media.

Ad esempio in figura dopo 25 abilitazioni della bobina viene calcolata la media eseguendo la divisione del registro 900 in cui è presente la somma delle 25 conversioni per il registro indice r(899) che sarà 25. Il risultato viene spostato su r(300).

• CONTATTI TASTI

I contatti tasti non sono stati modificati solo che per il PLC lilith hanno il seguente funzionamento: per compatibilita tra i PLC dei 32 tasti della tastiera del PLC lilith solo i tasti funzione F1...F8 sono impostabili in un programma gli altri vengono gestiti da sistema operativo per l'immissione delle variabili.

Quando si programma un PLC Lilith il dialog Box della impostazione dei tasti diviene il seguente: come si vede sono rappresentati i tasti da F1 a F8.

