

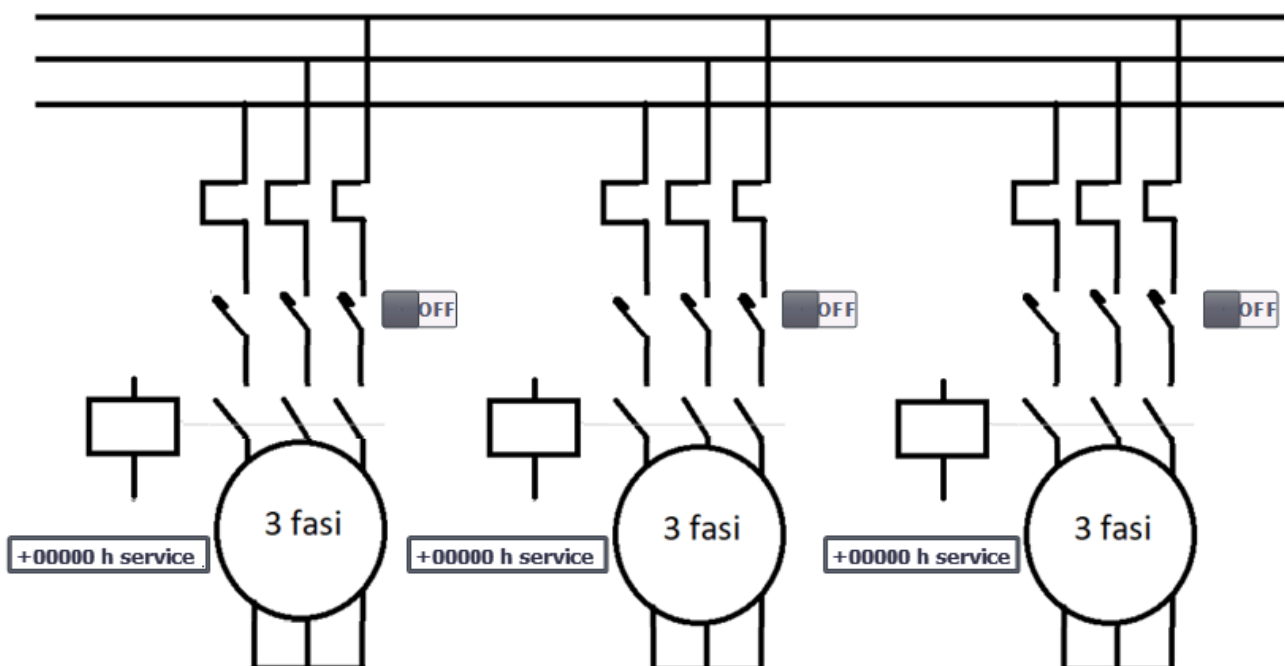
In quasi tutti gli ambiti delle automazioni industriali potrebbe essere necessario controllare dall'interfaccia HMI uno o più motori asincroni trifase.

Questi possono essere collegati a sistemi di sollevamento sia di liquidi tramite pompe che di materiali semilavorati tramite carriponte gru o semplici convogliatori, (conveyors).

In molti casi, ad esempio nelle applicazioni su sistemi di pompe, è necessario contare le ore di servizio cumulative.

Vediamo che questo può avvenire tramite la configurazione del blocchetto software RTM.

Lo schema elettrico sia quello mostrato in figura.



Le tre linee sono elettricamente identiche anche se nulla vieta che i tre motori o pompe siano di taglie diverse.

Ogni linea è sezionata dal proprio salvatore caratterizzato da un intervento di tipo magneto termico.

La commutazione di questo sezionatore sarà controllato dall'interruttore bistabile visibile affianco dei contatti con posizione naturale in OFF.

I motori sono controllati dai rispettivi teleruttori che chiudono le linee L1,L2,L3 sui morsetti U,V,W le cui bobine risultano collegate a stella senza possibilità di cambiare la configurazione a triangolo.

Il comando dei motori avverrà toccando il motore stesso sull'area grafica dell'HMI con un effetto Toggle ovvero al primo tocco si avvia e al secondo tocco di arresta.

I tre motori sono indipendenti e muniti ciascuno di un contatore cumulativo delle ore di servizio.

Ogni contatore è gestito dal comando RTM e in caso di spegnimento della macchina o messa in pausa riprenderà il conteggio orario dal momenti in cui è stato interrotto.

Solitamente i sistemi di pompe hanno bisogno di questo tipo di conteggio ore di servizio sia per determinare i consumi che per definire il momento in cui sarà necessario intervenire con le dovute manutenzioni.

Le parti sensibili di questa interfaccia HMI saranno i tre motori, in modalità toggle, e i tre switch dei salvamotore in modalità bistabile, mentre bobine e i contatti dei vari sezionatori sono liste grafiche che reagiscono allo stato dei comandi sopradetti.





Graphic lists	
Name ▲	Selection
bobina	Bit (0, 1)
motore	Value/Range
teleruttore	Bit (0, 1)
termico	Bit (0, 1)
<Add new>	

Quasi tutte le liste grafiche saranno abbinate a variabili a due stati quindi BOOL a parte quella che controlla la rotazione del motore che come vediamo nella prossima immagine rappresenta le bobine interne nelle varie situazioni angolari.

Ogni motore è costituito da una lista grafica contenete 4 immagini.

La prima è quella mostrata nella condizione di riposo, visibile nell'immagine, che contiene la scritta 3 fasi, le rimanenti 3 mostrano le bobine disegnate con un mutuo sfasamento di 120 gradi, e anche per ogni immagine una rotazione di 120 gradi.

Graphic lists	
Name ▲	Selection
bobina	Bit (0, 1)
motore	Value/Range

Graphic list entries				
Default	Value ▲	Graphic na...	Graphic	
<input type="radio"/>	0	mat spentp		
<input type="radio"/>	1	mat 1		
<input type="radio"/>	2	mat 2		
<input type="radio"/>	3	mat 3		

Alla selezione di zero nella variabile di controllo il motore viene visualizzato fermo.

Quando si avvia si crea una rampa, sfruttano l'uscita di un timer TON con auto riavvio in grado di selezionare le immagini indicate con 1, 2, 3.

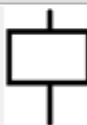
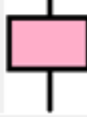
Quando il motore è in marcia si deve escludere dalla selezione l'immagine indicata con 0 per evitare un fastidioso sfarfallio.

Questo si ottiene facilmente sfruttando i comparatori IN_RANGE

Le bobine sono liste grafiche che rappresentano lo stato iniziale o non eccitata e lo stato finale, eccitata, in cui il contenuto del rettangolo viene colorato, in questo caso si è deciso che sia di colore rosa.

Va evitato il colore rosso perché indicherebbe una situazione di malfunzionamento o pericolo che in realtà non sussiste.

Graphic lists	
Name ▲	Selection
bobina	Bit (0, 1)
motore	Value/Range

Graphic list entries			
Value ▲	Graphic na...	Graphic	
0	contactor 0		
1	contactor 1		



Le altre liste grafiche sono:

La bobina dei teleruttori impiegati.

Si nota che allo stato zero della bobina di controllo è rappresentata da un rettangolo vuoto mentre allo stato eccitato la bobina è internamente colorata, in questo esempio di colore rosa.



Seguono le lista grafiche dei teleruttore e dello scatto di protezione termica.

Graphic lists	
Name ▲	Selection
teleruttore	Bit (0, 1)
termico	Bit (0, 1)

Graphic list entries			
Value ▲	Graphic na...	Graphic	
0	teleruttore ...		
1	teleruttore ...		

Sotto la lista grafica che anima il movimento dello scatto di protezione termica.

Graphic lists	
Name ▲	Selection
teleruttore	Bit (0, 1)
termico	Bit (0, 1)

Graphic list entries			
Value ▲	Graphic na...	Graphic	
0	termico off		
1	termico on		

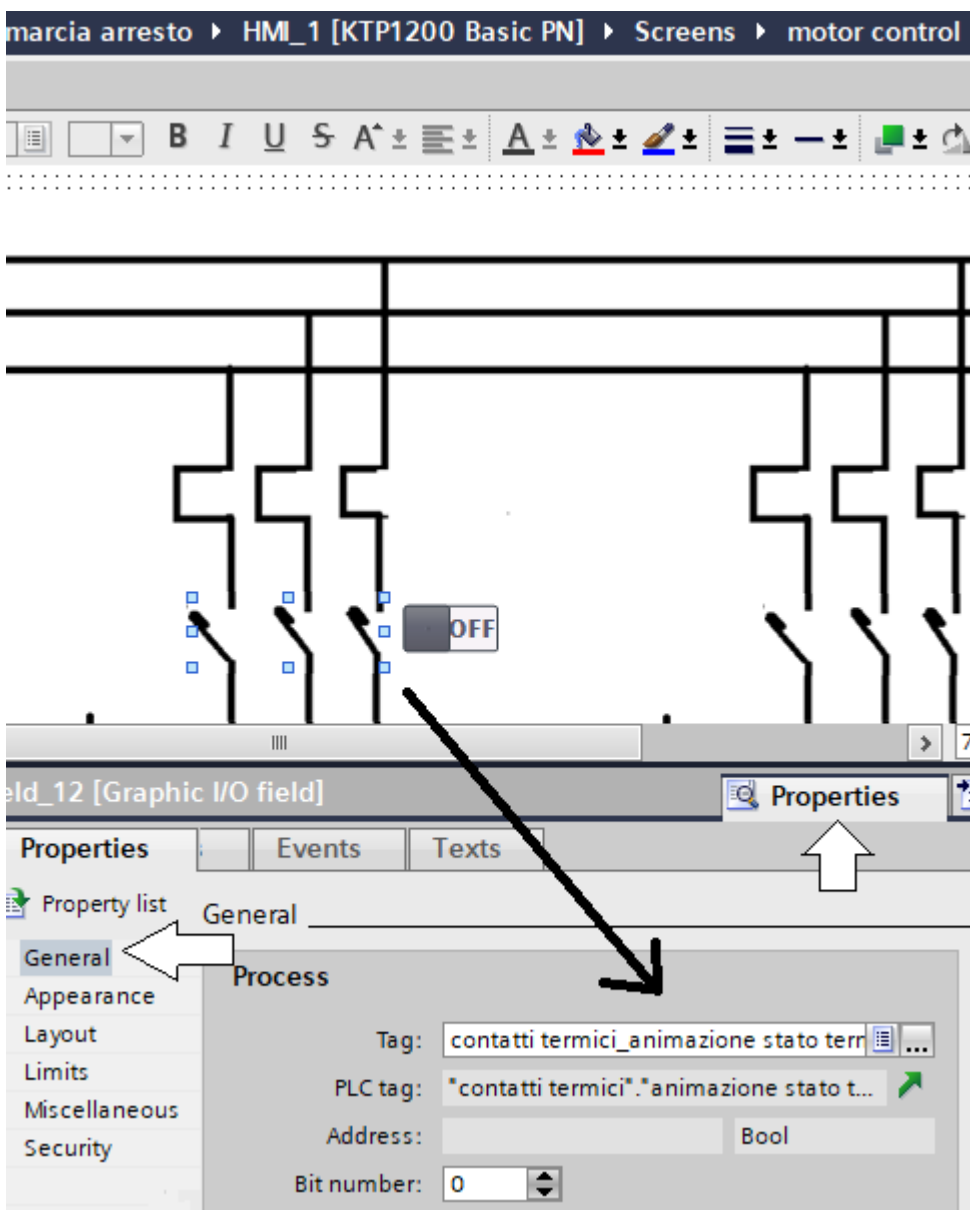
Le liste grafiche così costruite verranno ora copiate e incollate più volte in funzione del numero di esemplari da rappresentare nel pannello.

In sostanza abbiamo disegnato un motore ma lo useremo tre volte abbinandolo a tre variabili di controllo booleane diverse nel DB globale.

Ora dobbiamo predisporre le variabili, quindi si procede definendole nei DB di competenza, in questo caso decidiamo di porre tutte le variabili degli interventi termici in DB6.

marcia arresto ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ contatti termici [DB6]					
	Name	Data type	Start value	Retain	Ac
1	▼ Static			<input type="checkbox"/>	
2	animazione stato termico mot 1	Bool	false	<input type="checkbox"/>	
3	animazione stato termico mot 2	Bool	false	<input type="checkbox"/>	
4	animazione stato termico mot 3	Bool	false	<input type="checkbox"/>	
5	Comando ausiliare termico mot 1	Bool	false	<input type="checkbox"/>	
6	Comando ausiliare termico mot 2	Bool	false	<input type="checkbox"/>	
7	Comando ausiliare termico mot 3	Bool	false	<input type="checkbox"/>	

L'associazione delle variabili avviene in proprietà della lista grafica quando posta nell'area del pannello in cui si vuole che appaia.

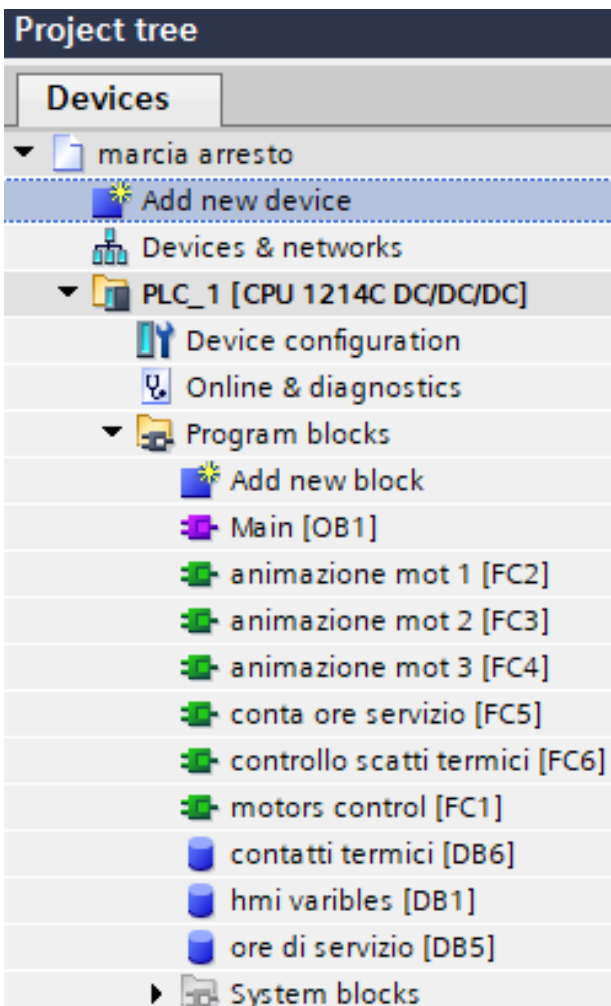


Con la stessa metodologia si dovranno associare tutte le altre tags ai rispettivi elementi grafici compresi i motori che risulteranno essere oggetti composti. La composizione consiste nella sovrapposizione di due elementi, il primo è il pulsante nascosto o meglio reso trasparente e il secondo le immagini della lista grafica.

Nel DB1, di tipo globale, metteremo tutte le variabile di interscambio tra PLC e HMI, come nell'immagine.

marcia arresto ▶ PLC_1 [CPU1214C DC/DC/DC] ▶ Program blocks ▶ hmi variables [DB1]					
	Name	Data type	Start value	Retain	A
1	▼ Static			<input type="checkbox"/>	
2	mot 1 start/stop	Bool	false	<input type="checkbox"/>	
3	mot 2 start/stop	Bool	false	<input type="checkbox"/>	
4	mot 3 start/stop	Bool	false	<input type="checkbox"/>	
5	anim mot 1	Int	0	<input type="checkbox"/>	
6	anim mot 2	Int	0	<input type="checkbox"/>	
7	anim mot 3	Int	0	<input type="checkbox"/>	
8	rampa animazione m...	DInt	0	<input type="checkbox"/>	
9	rampa animazione m...	DInt	0	<input type="checkbox"/>	
10	rampa animazione m...	DInt	0	<input type="checkbox"/>	
11	Animazione teleruttur...	Bool	false	<input type="checkbox"/>	
12	Animazione teleruttur...	Bool	false	<input type="checkbox"/>	
13	Animazione teleruttur...	Bool	false	<input type="checkbox"/>	
14	animazione bobina 1	Bool	false	<input type="checkbox"/>	
15	animazione bobina 2	Bool	false	<input type="checkbox"/>	
16	animazione bobina 3	Bool	false	<input type="checkbox"/>	

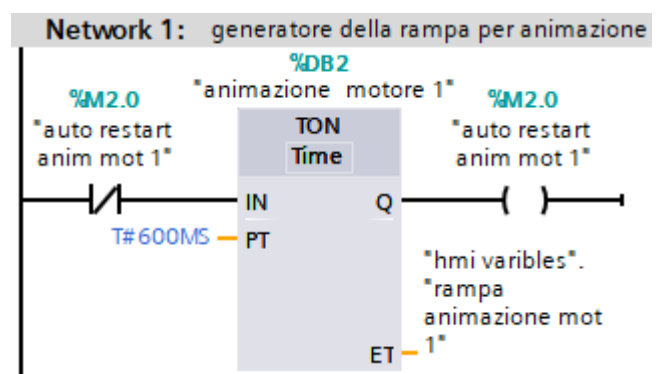
Osserviamo, qui sotto, il project tree consigliato per questa realizzazione:



Come è evidente si è preferito suddividere le animazioni e rispettivi controlli per i tre motori in gioco in tre diverse funzioni FC.

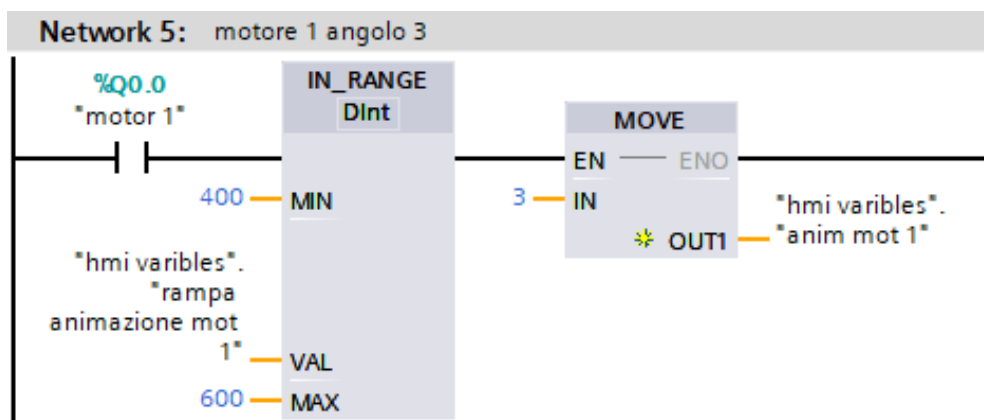
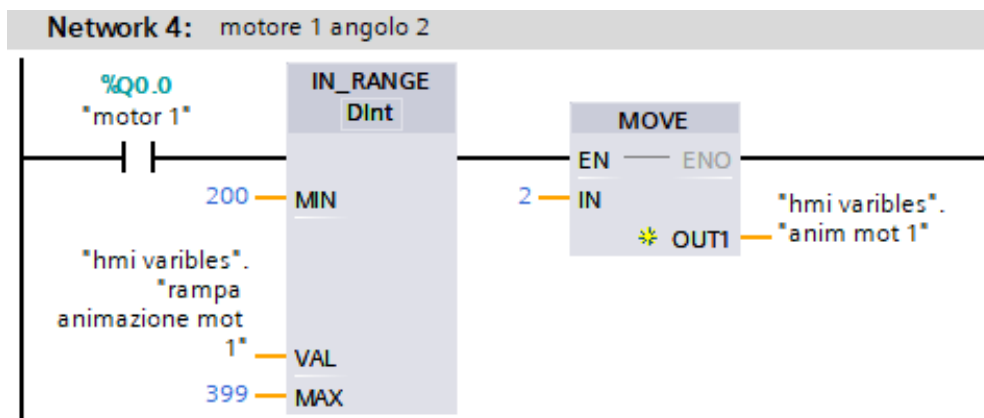
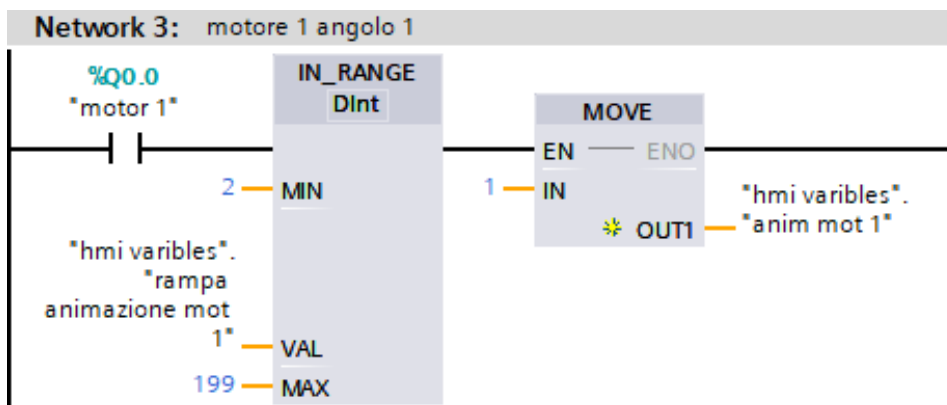
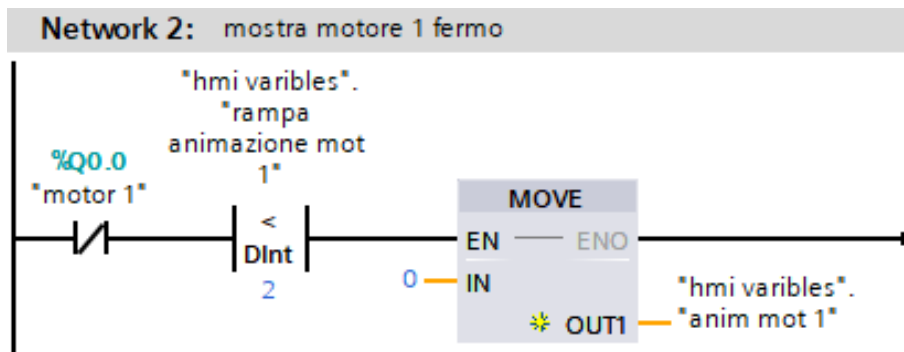
Nulla vieta, a vantaggio dell'eleganza di programmazione, di realizzare il programma usando un unico FB parametrizzato, tramite il block interface, su tre diverse istanze.

Ognuna delle funzioni FC2, FC3, FC4 contiene un generatore di onda a dente di segna e i rispettivi comparatori IN_RANGE per mostrare l'immagine della rispettiva lista grafica.



I tre timer devono essere indipendenti e quindi ricreati senza fare copia incolla per non conflituare a livello delle variabili contenute nel DB di istanza del timer TON.

I comparatori di tipo IN_RANGE saranno:

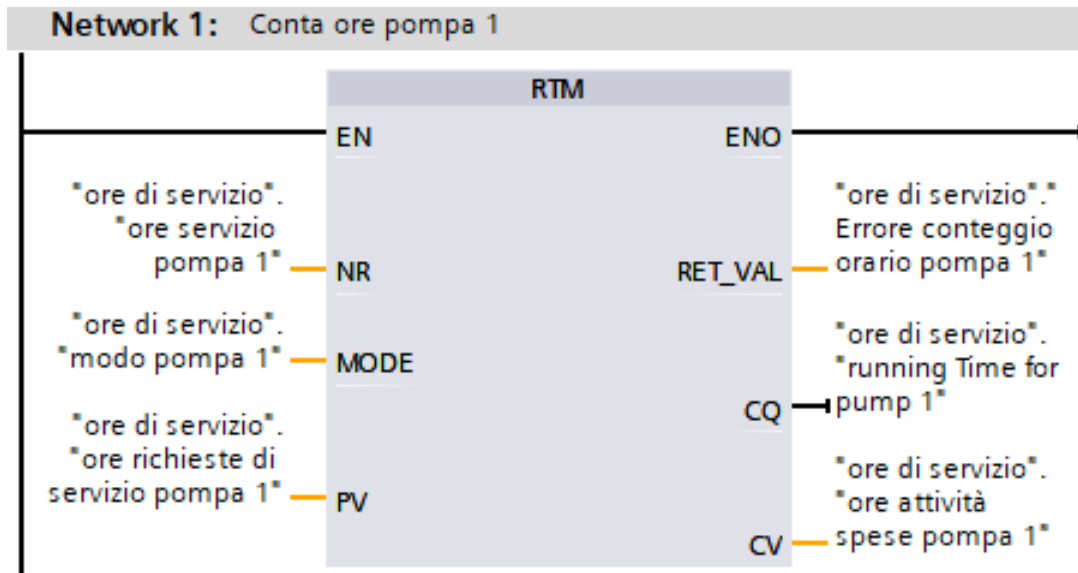


I contenuti di FC3 e FC4 sono logicamente (stessa struttura) uguali ma tipologicamente (indirizzi e etichette) diversi, per adeguarsi alle altre linee di motori da controllare.

Il conta ore di servizio.

Questa funzione viene realizzata facilmente in Step7 grazie alla presenza, nelle librerie, del comando RTM.

Il blocco funzione FC5 contiene tre segmenti contenenti ciascuno la rispettiva implementazione del comando RTM.



Il sistema è cumulativo e prosegue dal valore raggiunto qualora si dovesse tenere fermo uno dei motori per il tempo desiderato.

Le variabili sono contenute in un DB dedicati a questo scopo, DB5 -> conta ore di servizio.

marcia arresto ▶ PLC_1[CPU 1214C DC/DC/DC] ▶ Program blocks ▶ ore di servizio[DB5]					
	Name	Data type	Start value	Retain	Acc
1	Static			<input type="checkbox"/>	
2	ore servizio pompa 1	RTM	0	<input type="checkbox"/>	
3	ore servizio pompa 2	RTM	0	<input type="checkbox"/>	
4	ore servizio pompa 3	RTM	0	<input type="checkbox"/>	
5	ore richieste di servizio pompa 1	DInt	0	<input type="checkbox"/>	
6	ore richieste di servizio pompa 2	DInt	0	<input type="checkbox"/>	
7	ore richieste di servizio pompa 3	DInt	0	<input type="checkbox"/>	
8	modo pompa 1	Byte	16#0	<input type="checkbox"/>	
9	modo pompa 2	Byte	16#0	<input type="checkbox"/>	
10	modo pompa 3	Byte	16#0	<input type="checkbox"/>	
11	running Time for pump 1	Bool	false	<input type="checkbox"/>	
12	running Time for pump 2	Bool	false	<input type="checkbox"/>	
13	running Time for pump 3	Bool	false	<input type="checkbox"/>	
14	Errore conteggio orario pompa 1	Int	0	<input type="checkbox"/>	
15	Errore conteggio orario pompa 2	Int	0	<input type="checkbox"/>	
16	Errore conteggio orario pompa 3	Int	0	<input type="checkbox"/>	
17	ore attività spese pompa 1	DInt	0	<input type="checkbox"/>	
18	ore attività spese pompa 2	DInt	0	<input type="checkbox"/>	
19	ore attività spese pompa 3	DInt	0	<input type="checkbox"/>	

Le funzioni di conteggio orario della CPU avvengono su variabili a 32 bit, quindi con ampio range prima della saturazione.

Le variabili da parametrizzare sono:

Variabile nel blocchetto comando	Data type	descrizione
NR	RTM	Numero del contatore delle ore di funzionamento La numerazione inizia con 0. Per informazioni sul numero di contaore di esercizio della CPU, fare riferimento ai dati tecnici.
MODE	Byte	JOB ID: 0: lettura (lo stato viene quindi scritto in CQ e il valore corrente in CV). Dopo che il contatore delle ore di esercizio ha raggiunto $(2^{31}) - 1$ ora, si ferma al valore più alto visualizzabile ed emette un messaggio di errore "Overflow". 1: inizio (dall'ultimo valore del contatore) 2: fermati 4: impostato al valore specificato nel parametro PV 5: impostare al valore specificato nel parametro PV e avviare 6: impostare al valore specificato nel parametro PV e stop
PV	DINT	Nuovo valore per il contatore delle ore di esercizio
RET_VAL	INT	Se si verifica un errore durante l'esecuzione dell'istruzione, il valore restituito contiene un codice di errore.
CQ	BOOL	Stato del contatore delle ore di esercizio (1: in corso)
CV	DINT	Valore attuale del contatore delle ore di esercizio

Il blocco funzione **FC1** contiene la decodifica e l'esecuzione dei comandi Toggle per attivare i motori e le grafiche a pannello.

Nel primo segmento si è implementato anche l'intervento dello scatto termico.

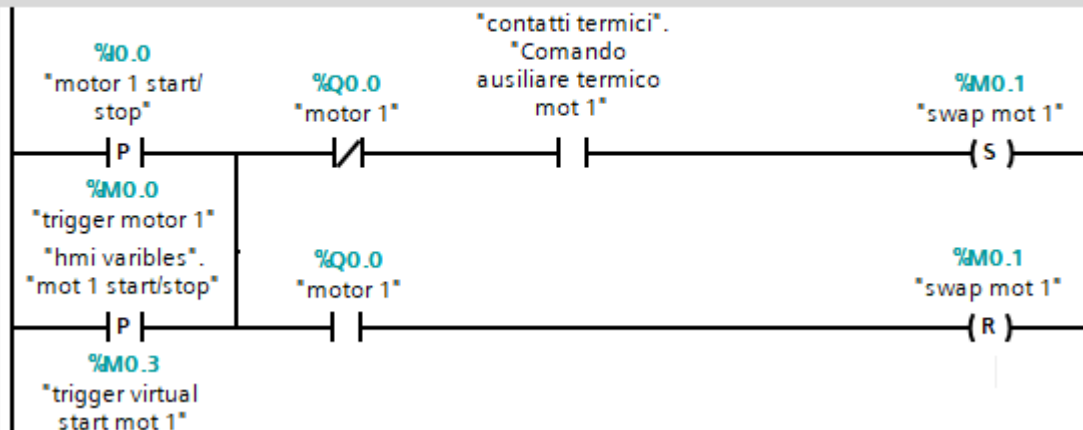
Nel secondo segmento sono raccolti e pilotati tutti gli elementi che devono essere impostato a TRUE quando questo comando è stato azionato.

I comandi sono, l'uscita fisica del motore, dichiarata nella Tags Tabe, l'animazione dei contatti del teleruttore e l'animazione del solenoide del teleruttore che andiamo a commutare.

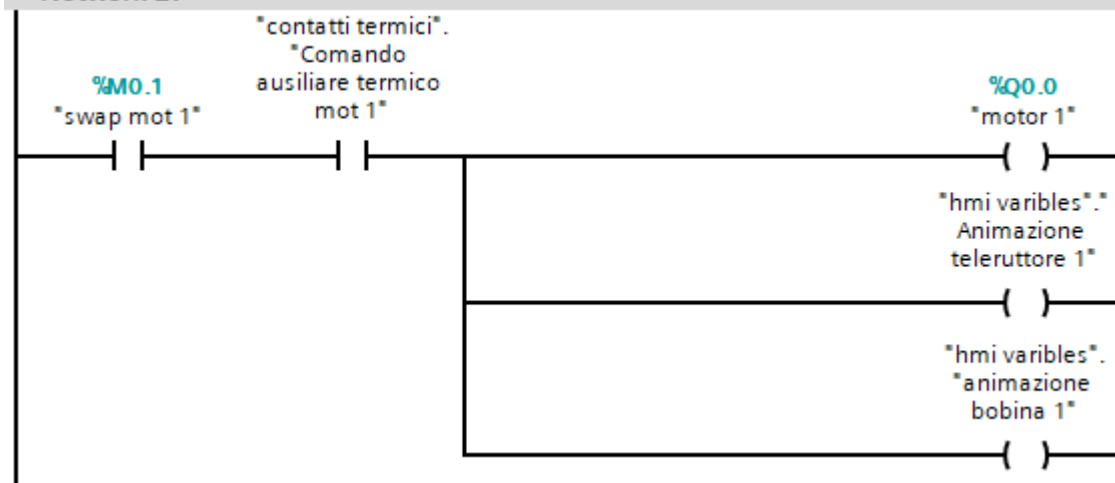
Nelle prossime immagini vediamo il contenuto del segmento 1 e 2 dell'FC1.

marcia arresto ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ motors control [FC1]

Network 1: set or reset for motor 1



Network 2: activate contactor motor 1

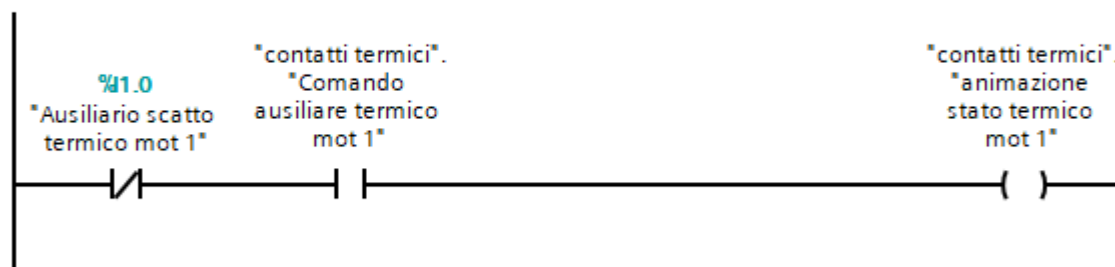


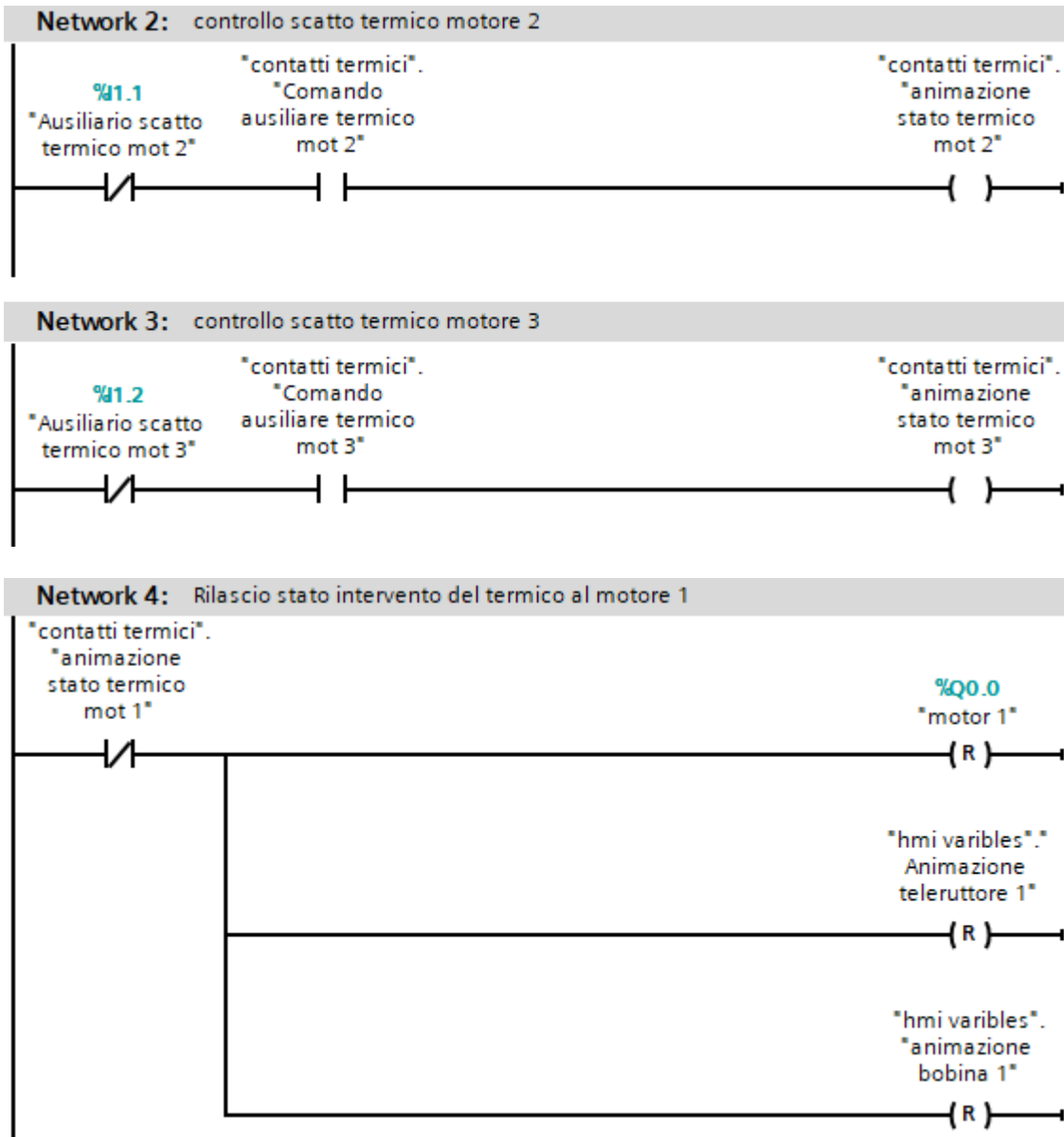
I due segmenti mostrati sopra vengono ripetuti, con il dovuto offset di indirizzi, per i rimanenti due motori e i relativi teleruttori.

L'intervento del termico è gestito nel blocco funzione FC6 in questa maniera.

marcia arresto ▶ PLC_1 [CPU 1214C DC/DC/DC] ▶ Program blocks ▶ controllo scatti termici [FC6]

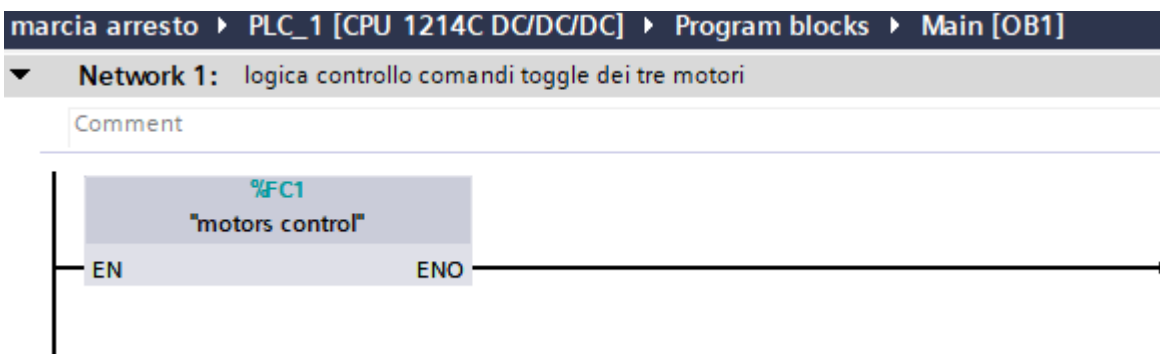
Network 1: controllo scatto termico motore 1

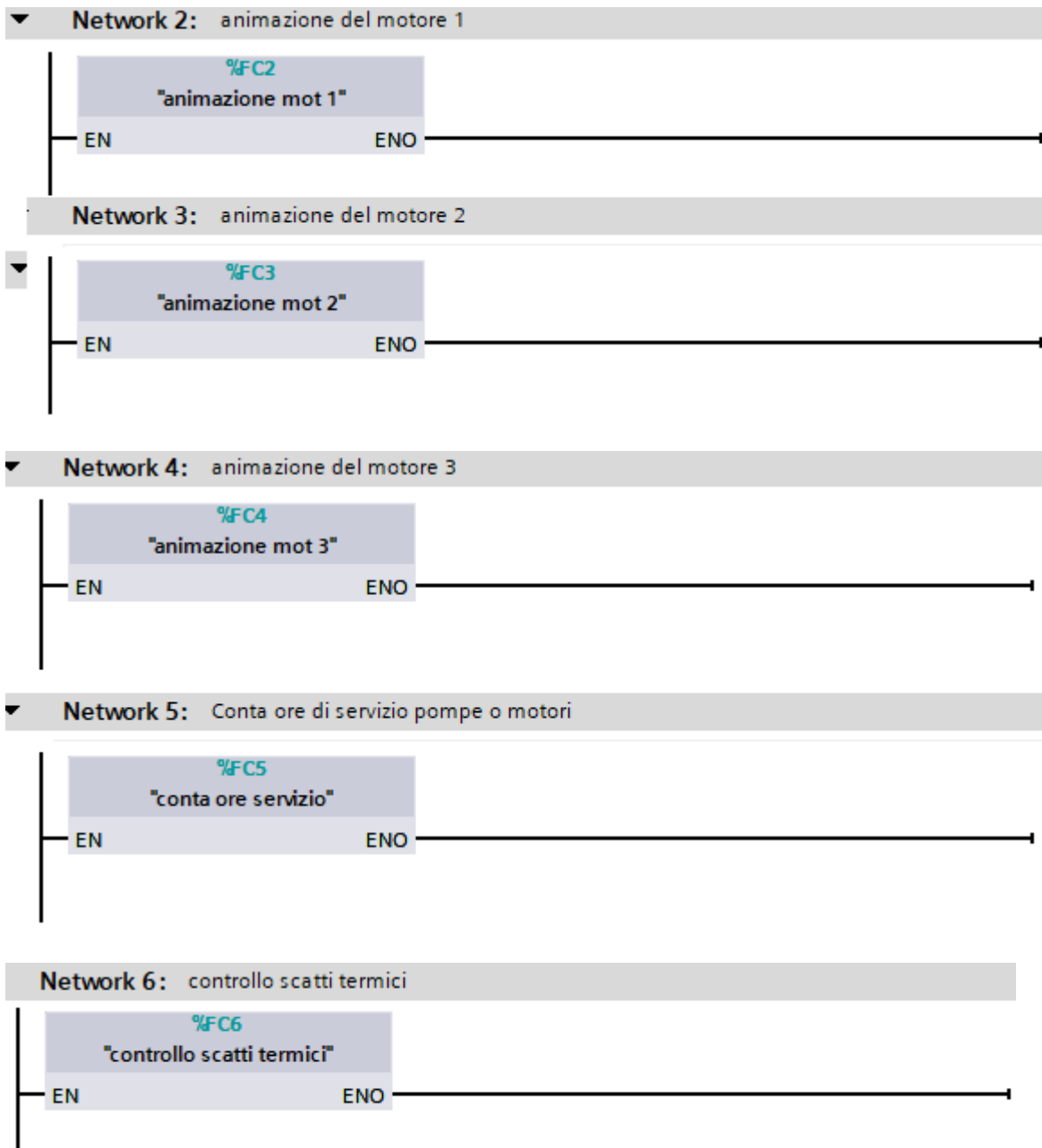




Il main program OB1

Nel blocco principale Main OB1 sono richiamati in maniera incondizionata e in sequenza tutti i blocchi funzione realizzati.





I sorgenti del programma sono disponibili nel sito web gtronic.it nella sezione programmazione di PLC, alla tabella Esercizi svolti, con titolo "Controllo pompe con motore trifase e cumulativo ore di servizio".

E' possibile prendere visione del video del funzionamento atteso si youtube al link:

<https://youtu.be/9ZWUhTRhOgQ>

Il libro di testo consigliato per apprendere le nozioni di base e poter realizzare questo progetto è:

Let's program a PLC First step in TIA Portal V17 Esercitazioni guidate per istituti tecnici: Terza edizione 2022, disponibile su Amazon, con ISBN 13: 979-8486336553

Editore prof. Gottardo