

Esercizio proposto: controllo inverter ing. Marco Gottardo.

I corsi di PLC con WinCC (Touch HMI) e SCADA (Movicom e ignition) si svolgono a **Padova** anche in modalità 3 week end, in caso di assenza forzata si recupera in altra data.

Via Austria 19/b, 35127, Zona Industriale

Chi abita a Padova può fare lezione serale lunedì e mercoledì dalle 20:00 alle 23:00, (attivazioni a ciclo continuo).

Il corso dura 40 e si rilascia un attestato di frequenza.

Costo compresa IVA 300€. Info e iscrizioni 335 6204892 (iscrizioni aperte tutto l'anno)

Libri a carico dello studente:

1. [Let's Program a PLC \(edizione 2016\).](#)
2. [Let's Program a PLC!!! Esercizi di programmazione dei PLC modelli S7300-400 e S7200 TIA Portal S7-1200 WinCC flexible per HMI](#)
3. [Advanced PLC programming](#)

Visita www.gtronic.it sezione programmazione PLC.

- Descrizione del problema
- Inverter
- Plc e collegamenti
- Programma di controllo
- Interfaccia HMI
- VARIANTI

Descrizione del problema

In questo esercizio si desidera controllare l'avviamento, la velocità e il senso di marcia di un motore elettrico tramite inverter. Il tutto verrà controllato tramite pannello operatore TP170 e gestito da PLC S7300 con CPU 314 IFM.

Inverter

In breve, è un dispositivo in grado di generare una corrente alternata a frequenza variabile da una corrente continua, o di variare la frequenza di alimentazione, per regolare la velocità di un motore elettrico. In questo esercizio viene impiegato un inverter da corrente alternata a corrente alternata (CA-CA) dove la tensione in entrata viene dapprima convertita in continua da un raddrizzatore e livellata, per poi generare le fasi, tramite oscillatore interni, applicate alla sezione di inversione. Lo scopo di questa doppia operazione è unicamente quello di variare la frequenza a piacere entro un intervallo prestabilito.

L'inverter, o variatore di velocità, utilizzato in questo esercizio è un Altivar 11 ATV11HU29M2E315, con alimentazione monofase 200/240V, per pilotare motori trifase 200/230V.

Tramite le linee di controllo L1 ... L4 è possibile impostare le modalità di funzionamento a quattro velocità prestabilite, o tramite un segnale analogico di ingresso, che può essere 0-5V nel caso si usi l'alimentazione interna dell'inverter o 0-10V, 0-20mA, 4-20mA, oppure X.Y mA, con X e Y impostabili, nel caso in cui il segnale analogico di ingresso sia alimentato esternamente, in questo caso il comune del segnale analogico deve essere collegato al comune dell'ingresso dell'inverter.

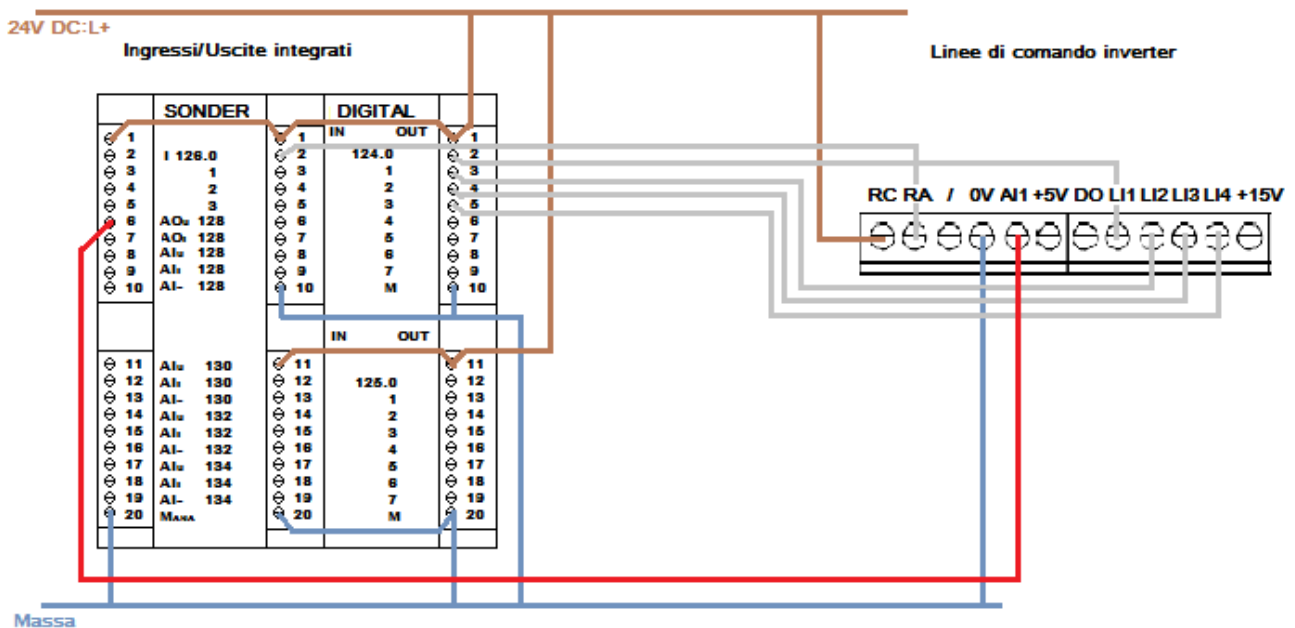
Plc e collegamenti

Il PLC utilizzato dispone di due byte di ingressi digitali, due byte di uscite digitali, quattro canali analogici di cui uno impostabile sia come ingresso che come uscita analogica, e gli altri solo ingressi analogici. Questo I/O integrato è più che sufficiente per il nostro esercizio.

Nello specifico, le linee di comando dell'inverter sono cablate nel modo seguente rispetto al PLC:

- RC Collegato a L+ del PLC
- RA Ingresso digitale 124.0, verifica se il contatto per la segnalazione di fault è aperto
- 0V Comune dal PLC
- AI1 Uscita analogica del PLC EW128
- +5V Non utilizzato
- DO Non utilizzato
- LI1 Uscita digitale 124.0, comando start avanti
- LI2 Uscita digitale 124.1, comando start indietro
- LI3 Uscita digitale 124.2, 0 velocità da canale analogico, 1 velocità preimpostata
- LI4 Uscita digitale 124.3, clear fault
- +15V Non utilizzato

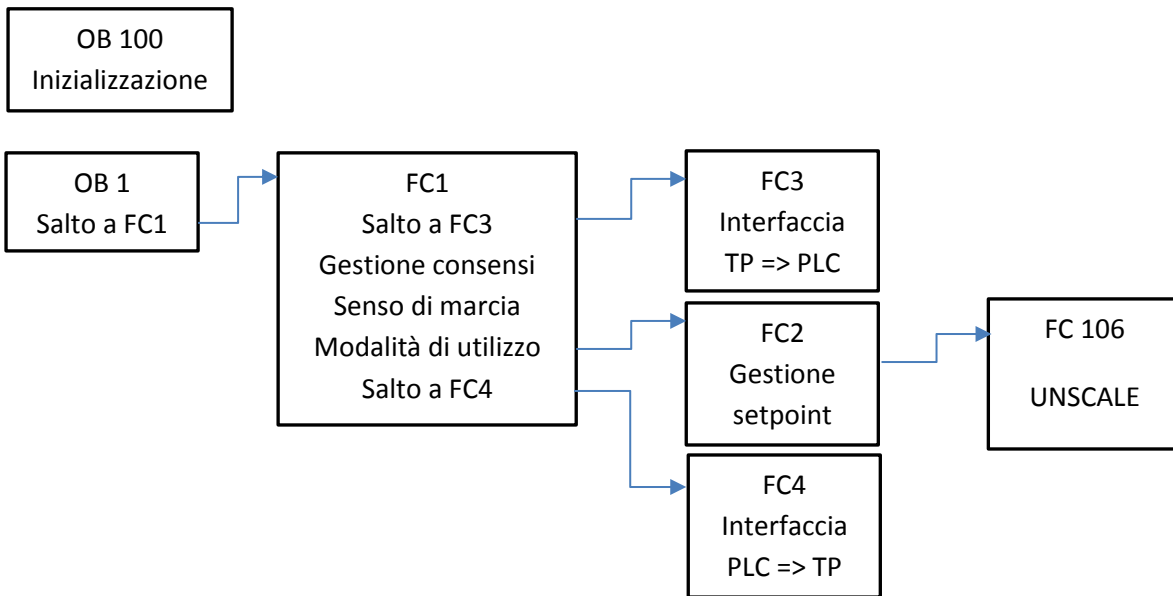
Il cablaggio complessivo seguirà il seguente schema:



Programma di controllo

Il programma gestisce lo start in avanti e indietro, selezionando la modalità di funzionamento, se con setpoint selezionato da utente o velocità fissa da inverter, con il vicolo di funzionamento con l'assenza di fault nell'inverter, e di cambio dei parametri in stop.

Il programma è strutturato secondo lo schema seguente:



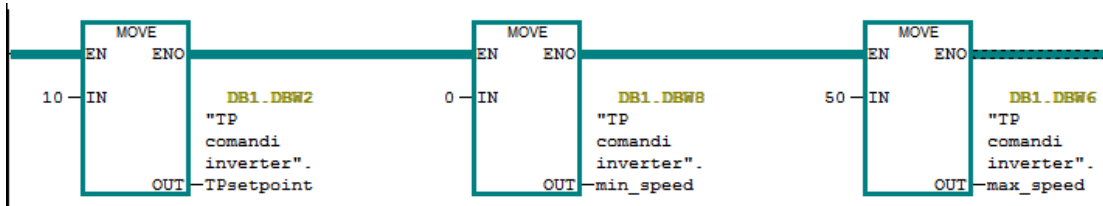
OB 100

È il blocco organizzativo che viene eseguito solo al passaggio del PLC da STOP a RUN. Qui si effettua l'inizializzazione del sistema, Nel primo segmento si impostano dei marker per la gestione dell'avvio. Nel secondo si impostano il valore di default del setpoint, il valore minimo e il valore massimo ammissibile.

Network 1

R	"start"	M2.3
S	"en_forward"	M2.0
S	"setpoint_mode"	M2.4

Network 2



FC 1

Il primo segmento è il salto a FC 3, per gestire l'interfaccia dati di ingresso.

I segmenti 2, 3, 4, gestiscono il consenso/mancato consenso alla partenza, e il riarmo in caso di fault.

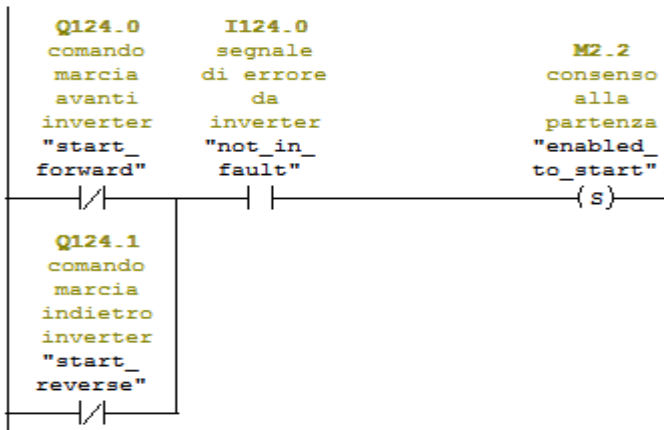
I segmenti 5, 6, gestiscono a marcia avanti / indietro.

I segmenti 7, 8, gestiscono le modalità di funzionamento.

Il segmento 9 è il salto a FC 4, per gestire l'interfaccia dati in uscita.

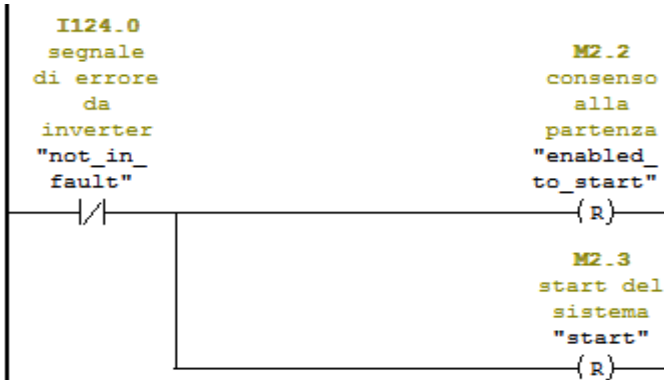
Le interfacce FC3 e FC4 sono state progettate per gestire la logica separandola da i comandi utente, o/e bordo macchina. Si possono quindi aggiungere nuove modalità di comando (pulsantiera, sensori, display) senza variare la logica, gestita nei segmenti dal 2 all' 8.

Network 2



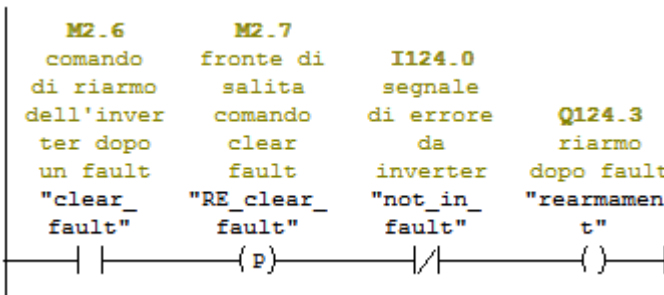
Se sono fermo (non sono ne in marcia avanti, ne in marcia indietro), e non sono presenti fault, ho il consenso a partire.

Network 3



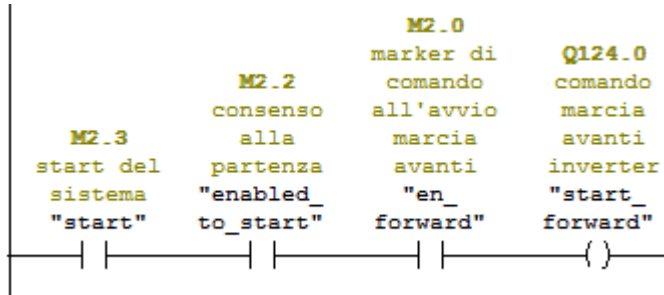
Se ho un fault, tolgo il permesso a partire e fermo il motore.

Network 4



Se ho un fault, il comando di riarmo genera l'impulso di riarmo da inviare all'inverter.

Network 5



Se ho il permesso a partire e ho impostato la marcia avanti, allo start invio all'inverter il comando di marcia avanti.

Network 6



Analogamente, se ho il permesso a partire e ho impostato la marcia indietro allo start invio all'inverter il comando di marcia indietro.

Network 7



Se non sono in modalità setpoint, passo automaticamente alla modalità velocità fissa, settando alto l'ingresso dell'inverter che gestisce queste due modalità (vedi inverter, LI3 = 1)

Network 8



Se sono in modalità setpoint, richiamo il blocco che gestisce il segnale analogico di uscita (FC 2), e l'ingresso dell'inverter che gestisce queste due modalità resta a zero (vedi inverter, LI3 = 0)

FC 2

L' FC 2 gestisce lo scalaggio del segnale analogico da grandezza ingegneristica al formato numerico da assegnare all'uscita analogica. Per fare ciò si dovrà usare la funzione FC 106 (UNSCALE) fornita dalle librerie Siemens, preparando i dati in modo opportuno per usare questa funzione.

la funzione UNSCALE richiede in ingresso dei parametri in formato Real, ovvero il setpoint, il valore minimo e il valore massimo, e un bool che indica se l'intervallo di uscita deve essere considerato unipolare (solo maggiore di zero) o bipolare (maggiore e minore).

In uscita invece la funzione ritorna un intero, ovvero il valore convertito, pronto per essere inviato all'uscita analogica, e una variabile Word che indica se ci son stati errori nella conversione.

Per usare la funzione UNSCALE prima di tutto si dovranno convertire i dati in reali, per poi essere utilizzati tramite variabili temporanee:

```
L "setpoint"  
ITD*  
T #x  
L "TP comandi inverter".min_speed  
ITD  
T #x0  
L "TP comandi inverter".max_speed  
ITD  
T #x1
```

*ITD : Integer To Double

Successivamente verrà richiamata la funzione UNSCALE:

```
CALL "UNSCALE"  
IN      :=#x  
HI_LIM  :=#x1  
LO_LIM  :=#x0  
BIPOLAR :=FALSE  
RET_VAL :=#y  
OUT     :=PQW128
```

Dove #x,#x0,#x1 sono le variabili temporanee di tipo Real, mentre #y è la variabile temporanea di tipo Word.

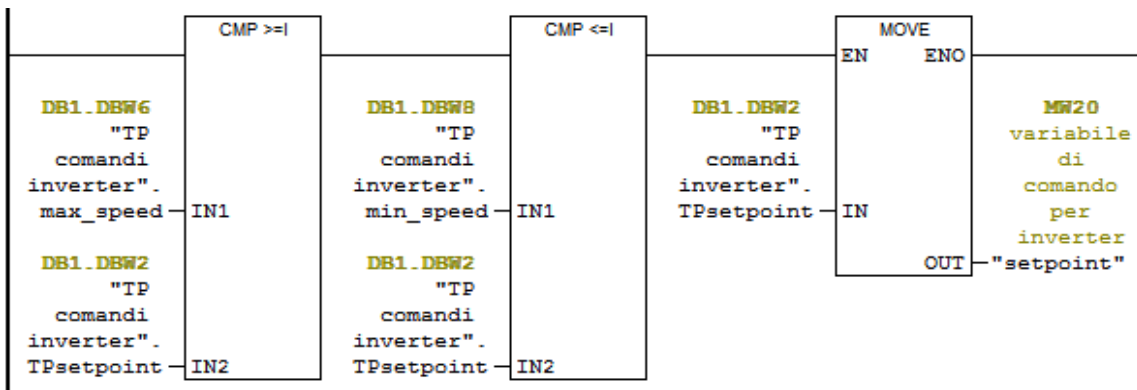
FC 3

L' FC 3 assegna i comandi dal mondo esterno, in questo caso solamente dall'HMI, alle variabili usate nella logica del programma.

Nello specifico assegna:

- "TP comandi inverter".TPstart (DB1.DBX 4.0) a "start" (M2.3)
- "TP comandi inverter".TPavanti (DB1.DBX 4.1) a "en_forward" (M2.0)
- "TP comandi inverter".TPcomando_setpoint (DB1.DBX 4.2) a "setpoint_mode" (M2.4)
- "TP comandi inverter".TPriarmo (DB1.DBX 4.3) a "clear_fault" (M2.6)

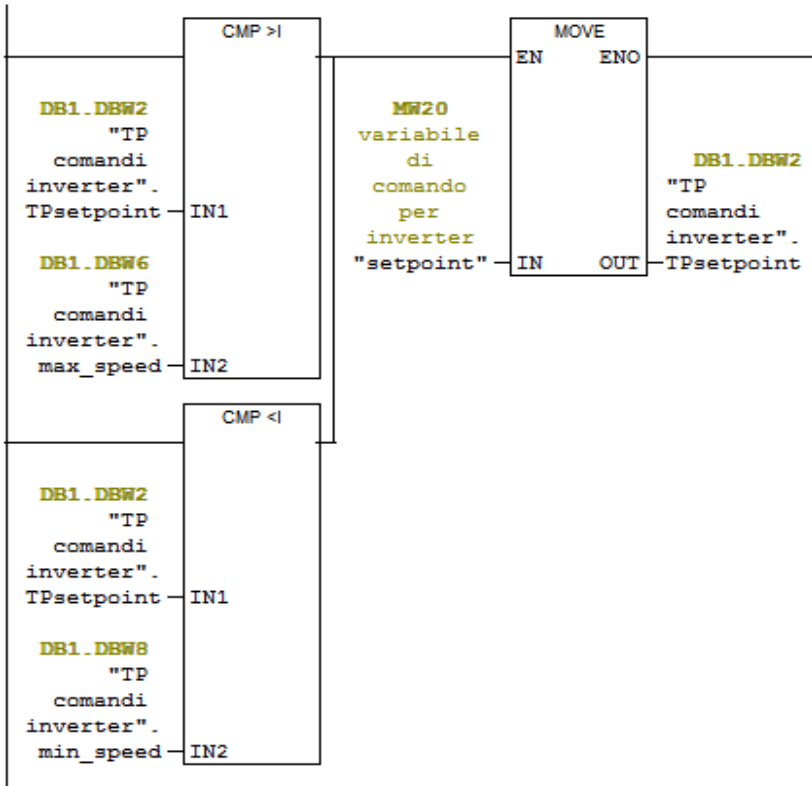
Inoltre c'è un segmento che assegna alla variabile "setpoint" il valore "TP comandi inverter".TPsetpoint, solo se "TP comandi inverter".TPsetpoint è compresa nei limiti imposti da "TP comandi inverter".max_speed e "TP comandi inverter".min_speed:



FC 4

Dualmente alla FC 3, l'FC 4 si occupa di scrivere i dati per il mondo esterno, in questo caso solo l'HMI, nello specifico:

Network 2:



Se il valore immesso nel TP per il nuovo setpoint è maggiore del massimo, o minore del minimo, il segmento cancella il dato immesso e lo sostituisce con l'ultimo dato valido

Network 3



In caso di fault, si segnala la presenza di fault al TP, e si resetta il comando di start.

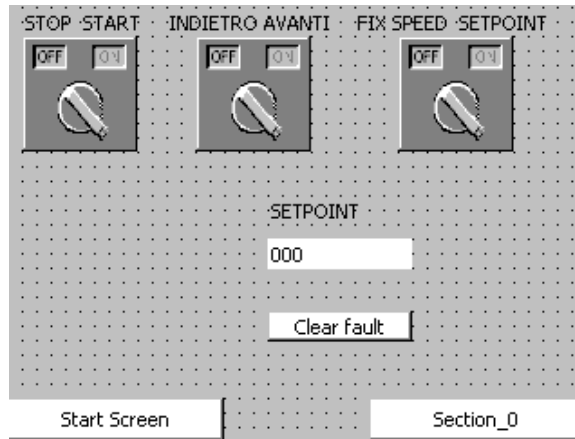
Con questo segmento si comunica il consenso allo start al TP

Interfaccia HMI

Il pannello operatore usato in quest'esercizio è un TP170. Questo pannello permette di effettuare le operazioni basilari per la comunicazione all'utente.

In questo pannello sono stati usati tre Switch per lo start/stop, avanti/indietro, setpoint/fixspeed, un IO field per scrivere/leggere il setpoint, e un pulsante per il clear fault.

Inoltre è stata impostata una finestra di errore che compare quando è presente un fault.



I tags per la gestione del display sono stati impostati come da immagine:

Name	Connection	Data type	Address	Array elements	Acquisition cycle	Comment
plc1_setpoint	Connection_1	Int	DB 1 DBW 2	1	1 s	
plc1_trigger_word	Connection_1	Int	DB 1 DBW 4	1	1 s	si usa solo il bit del fault, ovvero il bit 12
plc1_start	Connection_1	Bool	DB 1 DBX 4.0	1	1 s	
plc1_avanti	Connection_1	Bool	DB 1 DBX 4.1	1	1 s	
plc1_comando_setpoint	Connection_1	Bool	DB 1 DBX 4.2	1	1 s	
plc1_riarmo	Connection_1	Bool	DB 1 DBX 4.3	1	1 s	
plc1_fault	Connection_1	Bool	DB 1 DBX 4.4	1	1 s	
plc1_start_enabled	Connection_1	Bool	DB 1 DBX 4.5	1	1 s	

I tags rispecchiano la configurazione del DB del programma nel PLC, se non per il tag "plc1_trigger_word", infatti questo tag server per identificare il bit "plc1_fault" come trigger bit per la segnalazione dell'errore.

La segnalazione dell'errore avviene identificando un trigger bit in un specifico tag di tipo word, e associandone un testo che verrà visualizzato quando il trigger bit passa da 0 a 1. Nell'impostazione del trigger bit si nota come anche nell'HMI si consideri la modalità big endian per la scrittura dei dati in memoria.

Text	Class	Number	Trigger Tag	Trigger bit	Trigger address
Inverter in fault	Errors	1	plc1_trigger_word	12	DB 1 DBX 4.4

Gli oggetti del TP sono stati impostati nel seguente modo:

Switch stop/start

- Process Tag: "plc1_start"
- Animation, Enabled Object, Tag: "plc1_start_enabled", object state: Disabled, range: 0

Switch indietro/avanti

- Process Tag: "plc1_avanti"
- Animation, Enabled Object, Tag: "plc1_start" , object state: Enabled, range: 0

Switch fixspeed/setpoint

- Process Tag: "plc1_comando_setpoint"
- Animation, Enabled Object, Tag: "plc1_start" , object state: Enabled, range: 0

I due switch indietro/avanti e fixspeed/setpoint sono stati settati in modo che vengano disabilitati quando lo switch stop/start è sulla posizione di start.

IO Field setpoint

- Process Tag: "plc1_setpoint"
- Animation, Enabled Object, Tag: "plc1_comando_setpoint" , object state: Disabled, range: 0

L'IO field è settato in modo che sia possibile scrivere solo se è impostata la modalità setpoint nello switch.

Button clear fault

- Evento Press/Release: SetBit/ResetBit: "plc1_riarmo"
- Animation, Enabled Object, Tag: "plc1_fault" , object state: Disabled, range: 0

Il pulsante è settato in modo che sia abilitato solo nel caso in cui sia presente un fault

VARIANTI

- In questo esercizio è stata usata solo un'interfaccia HMI per comunicare con l'utente. Si provi ad inserire nel programma un'altra modalità di comunicazione con l'utente (pulsanti, potenziometri display 7 segmenti, ecc)
- La funzione FC 106, anche se garantita dalla Siemens, comporta la conversione di interi in reali per ottenere uno scalaggio intero. Si studi uno scalaggio intero da sostituire alla FC106, oppure direttamente alla FC2