Università degli Studi di Bergamo

Automazione Industriale

Laboratorio
3

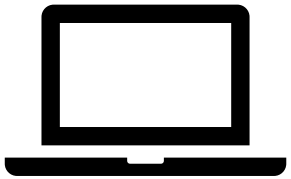
Configurazione HW

X20CP 1584

ETHERNET POWERLINK

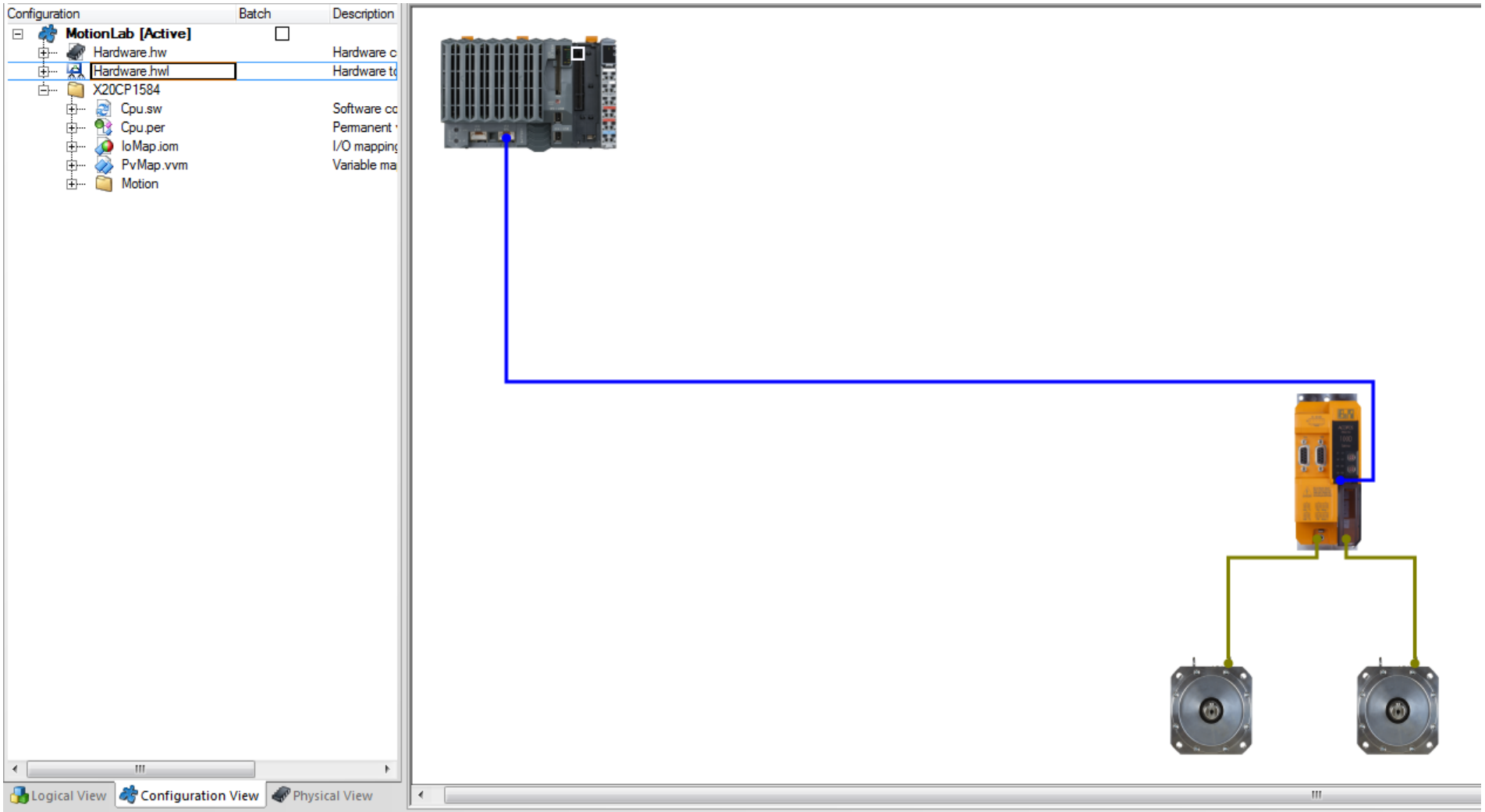
Driver

Ethernet



Motori sincroni

Configurazione HW



Configurazione Motori

Mot1 e Mot2 sono variabili di tipo MpAxisBasic

- Enable
- Power
- Home
- Stop
- MoveAbsolute
- MoveAdditive
- ReadyToPowerOn
- PowerOn
- IsHomed
- Position (attuale)
- Velocity (attuale)

Par_Mot1 e Par_Mot2 sono variabili di tipo MpAxisBasicParType

- Position
- Velocity
- Acceleration
- Home.Position

Configurazione Motori

Step per inizializzare il motore:

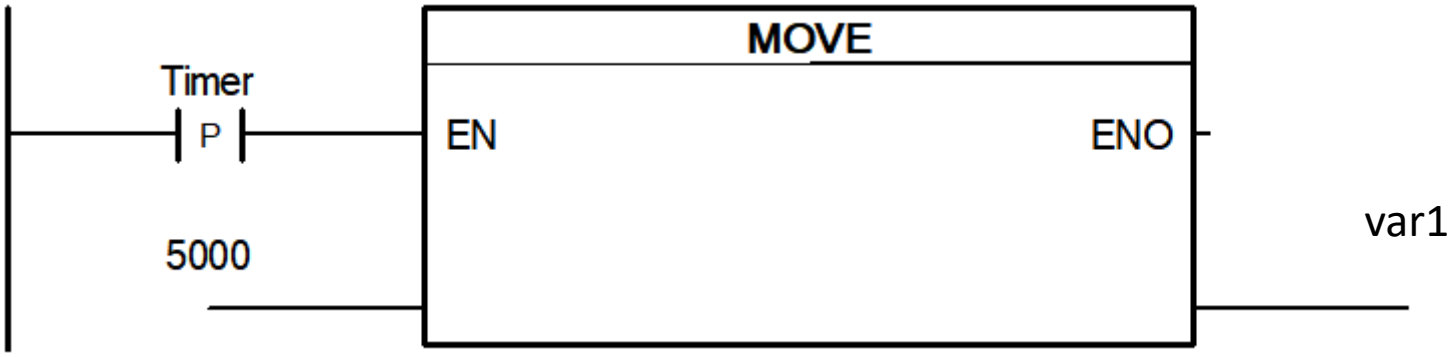
- Check ReadyToPowerOn
- Power
- Check PowerOn
- Home
- Check isHomed

Una volta inizializzato possiamo dargli i comandi di movimento

Configurazione Motori

If EN:

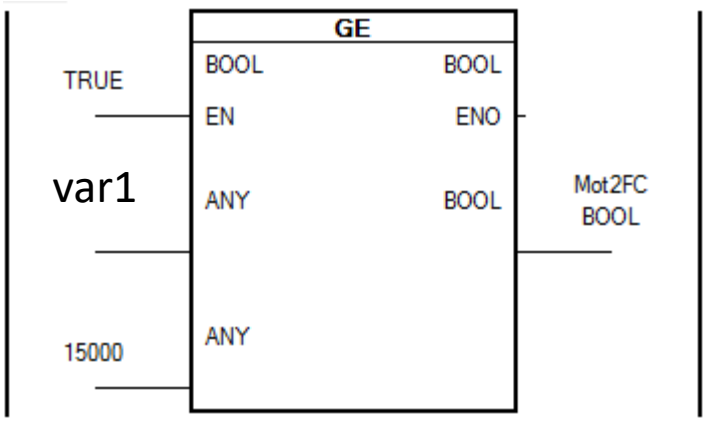
var1 = 5000



If EN:

if var1 >= 15000:

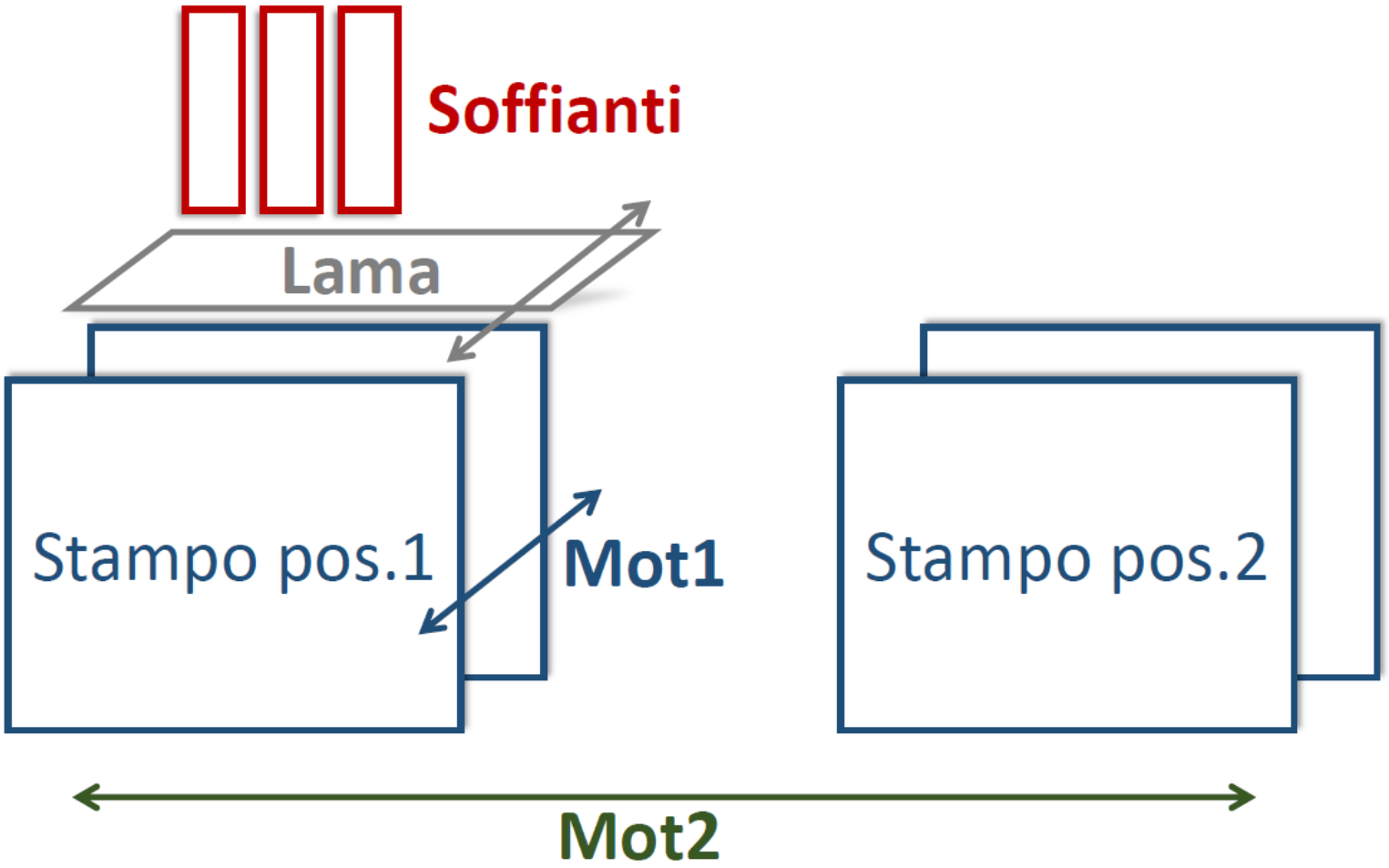
Mot2FC = True



Esercizio LADDER

Esercizio LADDER

Schema



Esercizio LADDER

Lo stampo (**Mot1**) si chiude seguendo due fasi:

- In una prima fase, l'avvicinamento viene eseguito in velocità, a 500 $unit/s$, fino al raggiungimento della posizione 237 $unit$

- Dalla posizione di 237 $unit$, lo stampo si deve avvicinare con una velocità limitata a 250 $unit/s$ comandato in posizione, fino al raggiungimento della posizione di chiusura a 359 $unit$.

Nei cambi di velocità si utilizzi sempre una accelerazione di 100 $unit/s2$

Esercizio LADDER

Una volta che lo stampo si è chiuso, la lama viene estesa per tagliare. Questa rimane estesa per 1 secondo, e quindi ritratta immediatamente.

A questo punto, l'asse che muove il sistema longitudinalmente (**Mot2**) sposta lo stampo da sinistra verso destra, alla velocità di 350 $unit/s$ fino al raggiungimento della posizione 2 a 359 $unit$.

Le soffianti vengono quindi attivate per avviare la fase di soffiaggio. Queste restano attive per 3 secondi, poi vengono disattivate.

Quindi, lo stampo viene aperto alla velocità di 500 $unit/s$, fino al raggiungimento della posizione 0 $unit$.

Nel frattempo, lo stampo viene riportato in posizione 1, con le stesse modalità con cui era stato portato in posizione 2.

Esercizio 1 - ST

Sistema che simula il comportamento di un motore dotato di protezione e allarme.

Premo il tasto di **avvio**:

- Se non vi è la protezione scatta **l'allarme**
- Se vi è la **protezione** il motore si avvia

Se durante l'allarme inserisco la protezione, l'allarme si ferma.

Il **motore** una volta avviato rimane acceso per 30 secondi. Se durante questi secondi viene premuto il tasto di **stop**, il motore viene fermato.

Esercizio 2 - ST

Sviluppare un FB che simuli il comportamento di un flip-flop set-reset

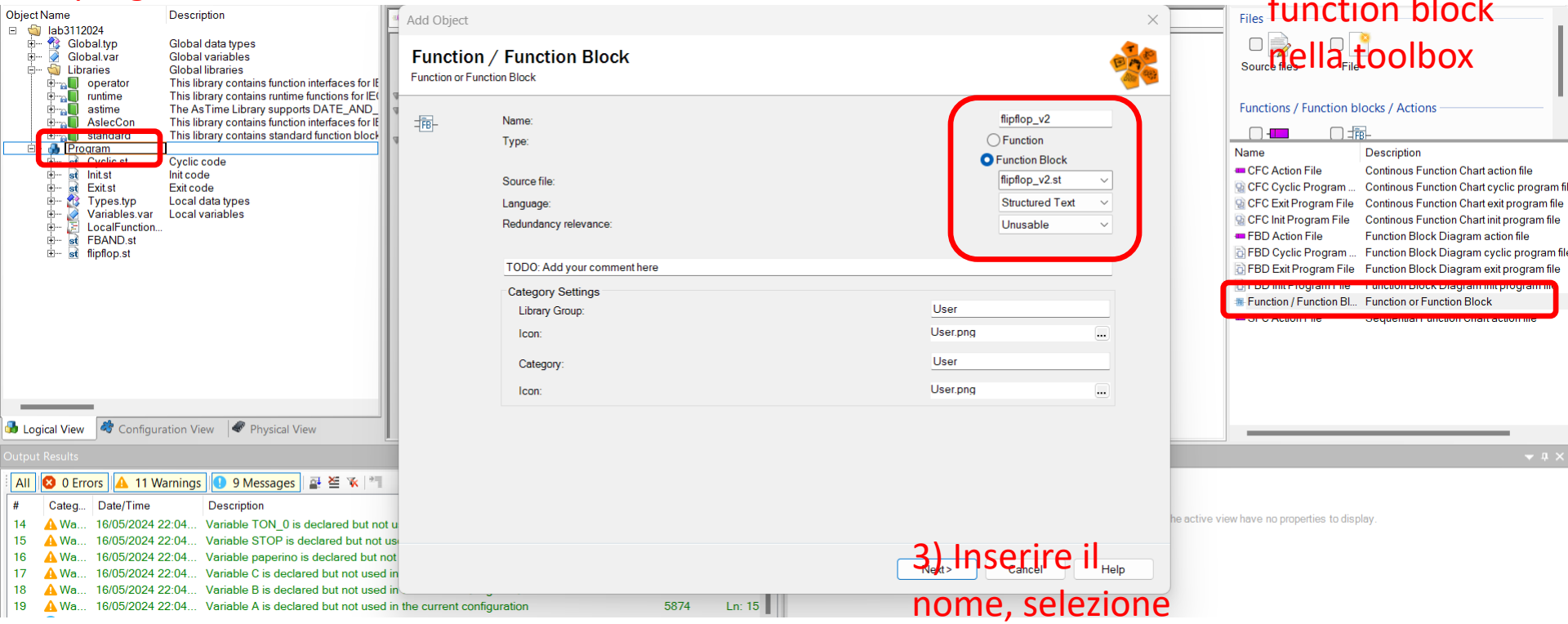
S	R	U
0	0	Uold
0	1	0
1	0	1
1	1	-

Sviluppare un FunctionBlock in ST che valuti gli stati dei seguenti segnali digitali tramite il flip-flop set-reset

Function block ST

1) click sul nome del programma

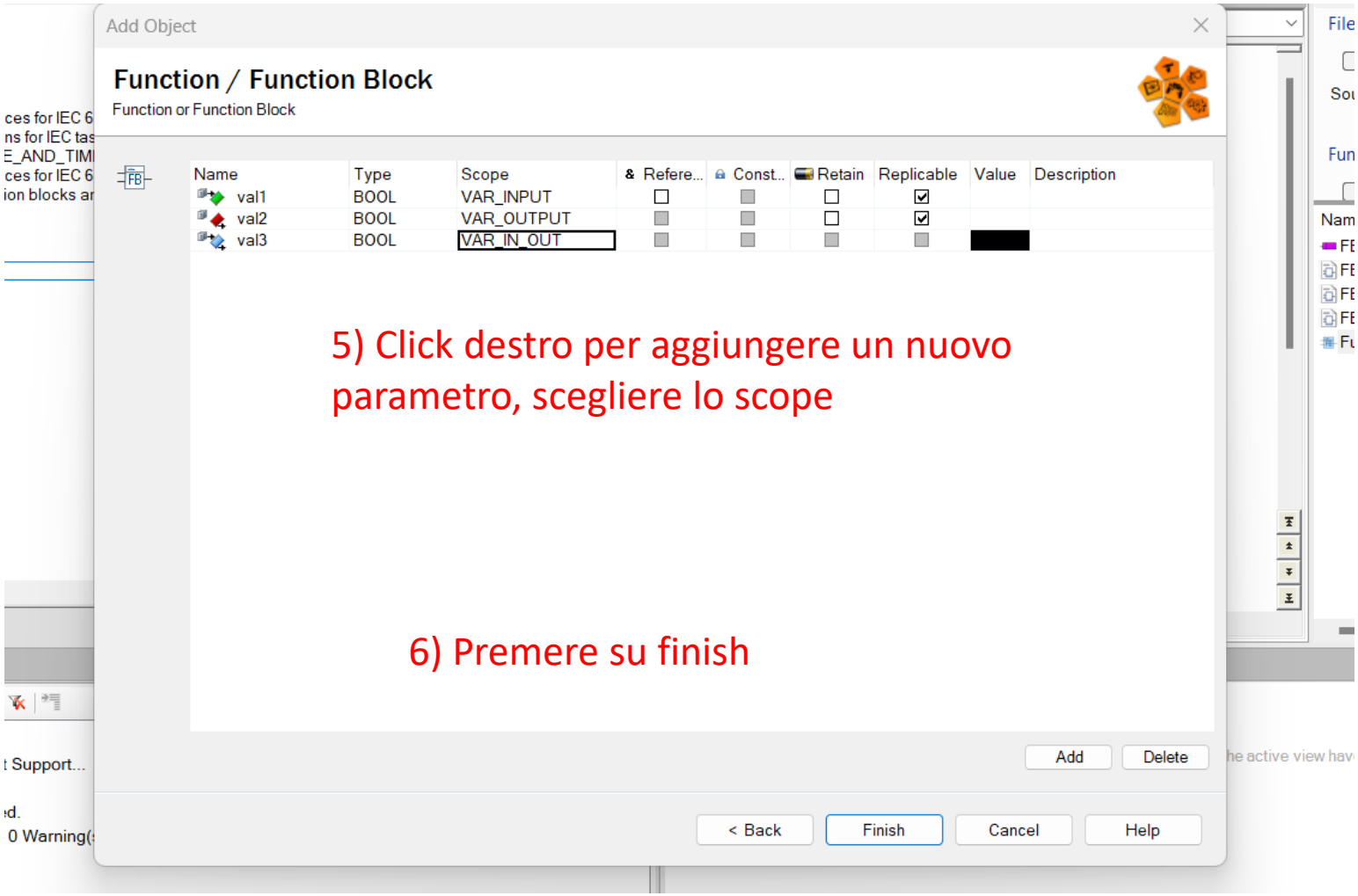
2) Cercare e selezionare il function block nella toolbox



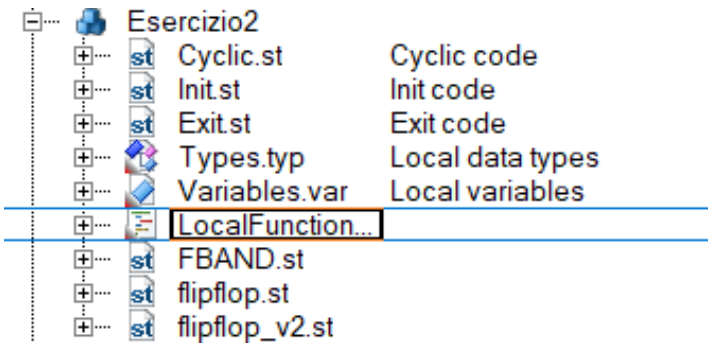
3) Inserire il nome, selezione Function Block come type e structured text come language

4) Premere Next

Function block ST

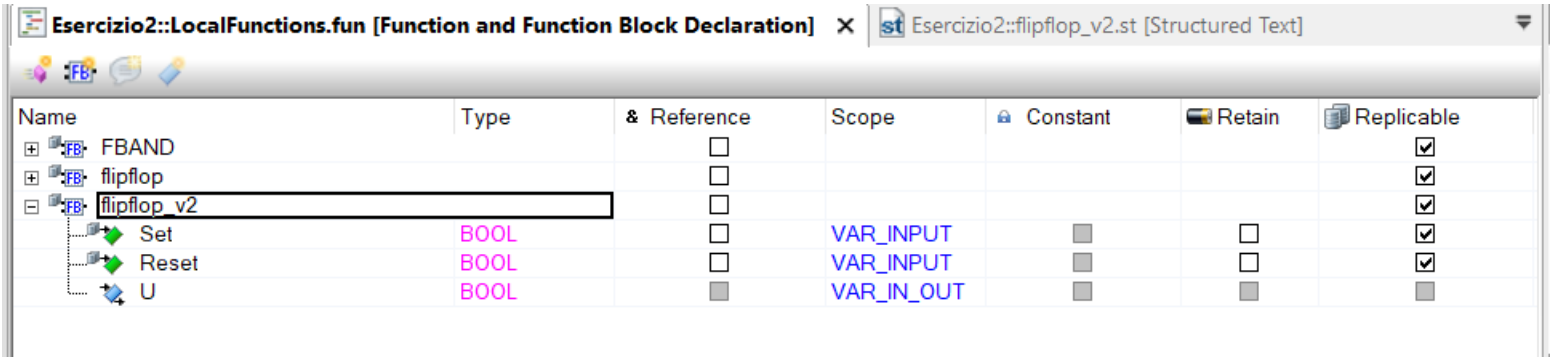


Function block ST

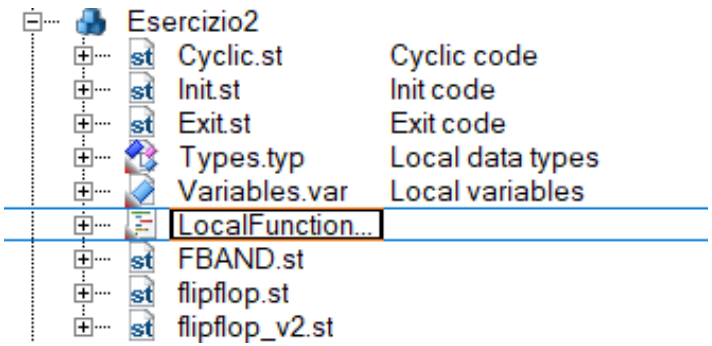


A sinistra avremo i function block con l'estensione del file .st

E un file LocalFunctions dove vengono mostrati i parametri/variabili dei FB (possono essere modificati

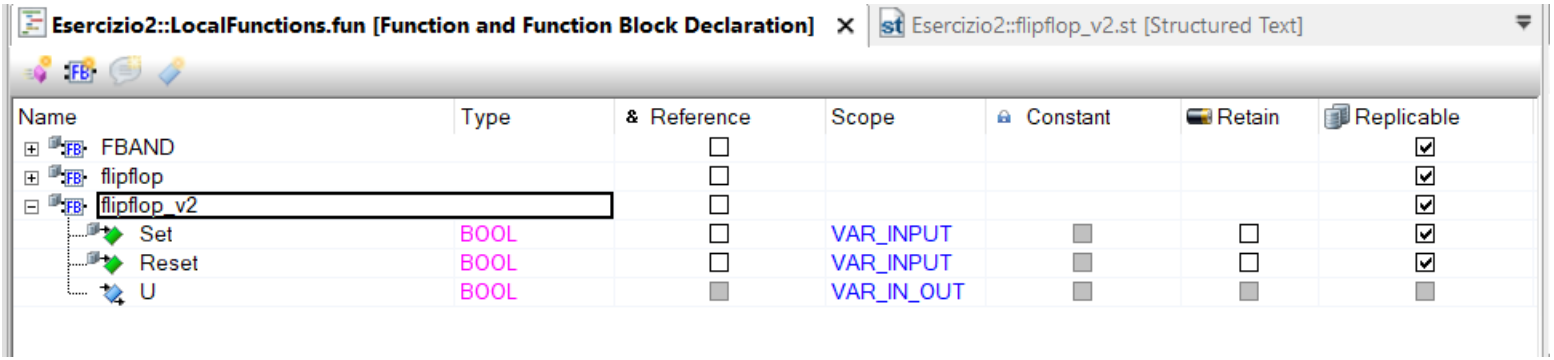


Function block ST



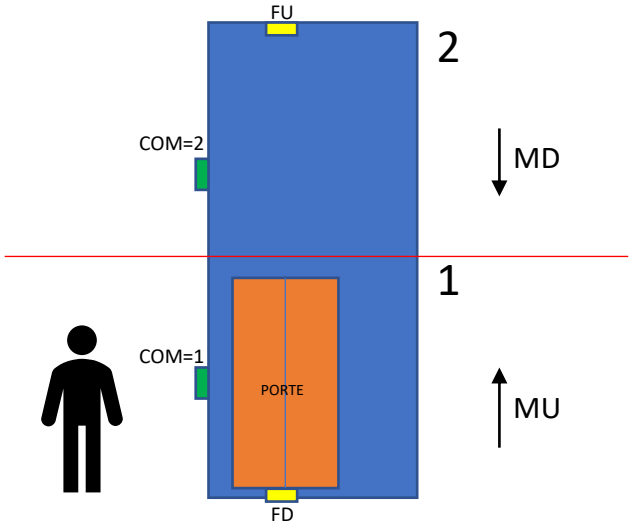
A sinistra avremo i function block con l'estensione del file .st

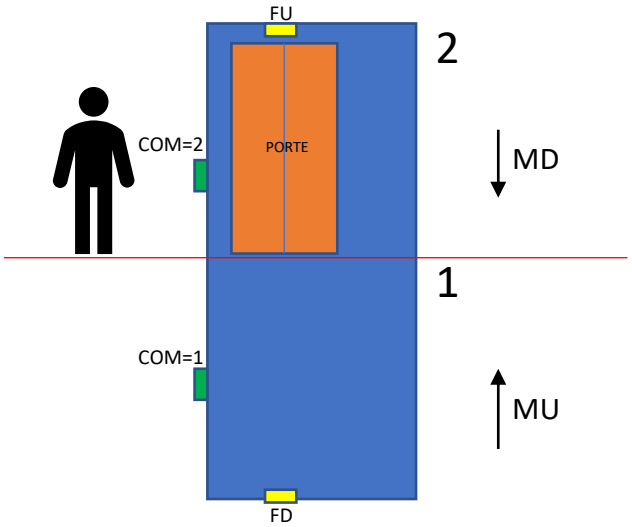
E un file LocalFunctions dove vengono mostrati i parametri/variabili dei FB (possono essere modificati



Esercizio 3 - ST

Sviluppo di un sistema di controllo di un ascensore.





L’ ascensore può andare solamente al primo e al secondo piano, i comandi vengono impartiti tramite i tasti COM.

Quando COM = 1 portare l'ascensore dal piano 1 al piano 2.

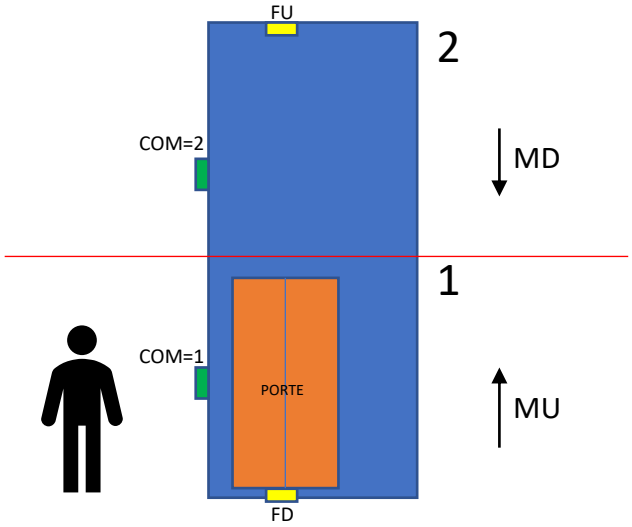
Quando COM = 2 portare l'ascensore dal piano 2 al piano 1.

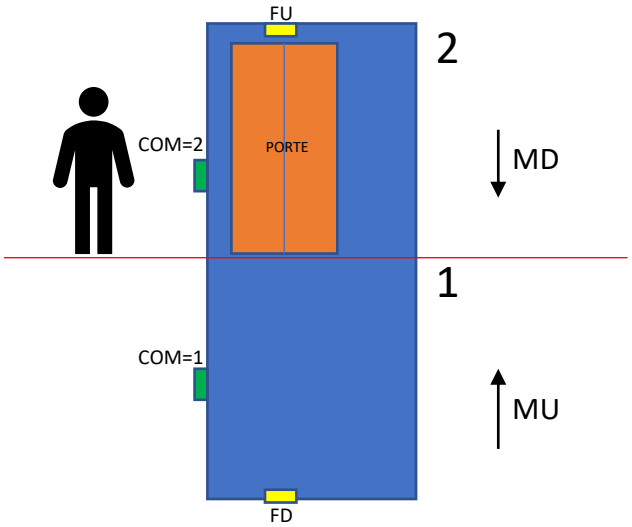
Una volta premuto il tasto **COM** il sistema si aspetta che le **PORTE** vengano chiuse, quindi viene attivato il motore (**MU** o **MD**) e una volta raggiunto uno dei fine corsa (**FU** o **FD**) vengono spenti i motori e aperte le porte.

Gestire il controllo della chiusura delle porte, dell’ avvio del motore, del raggiungimento del fine corsa e dell'apertura delle porte tramite Function Block.

Esercizio 3 - ST

Sviluppo di un sistema di controllo di un ascensore.





PORTE: 0 aperte, 1 chiuse (agire manualmente)

Non si deve gestire la chiamata dell'ascensore e quando si preme il tasto se l'ascensore non è nel tuo piano non succede nulla

La valorizzazione dei finecorsa FU e FD non è gestita, agire manualmente.

Inizialmente COM è uguale a 0

Suggerimento FB:

FINECORSA VAR_IN

PORTE VAR_IN_OUT

AZIONAMENTO VAR_OUT

Esercizio 4 – ST (BONUS EXE)

https://www.youtube.com/watch?v=m6zxS1-nhQw

Esercizio 4 – ST (BONUS EXE)

Sviluppare in Testo Strutturato il software di controllo di una linea automatica Cosberg.

La linea ha 5 stazioni di lavorazione consecutive collegate da un nastro trasportatore + pallet su cui viaggerà un pezzo dalla stazione 1 alla 5. Ogni stazione ha una fotocellula rappresentata da un ingresso digitale chiamato STAZ_#, dove # è il numero della stazione (quindi STAZ_1, …. STAZ_5).
Il nastro trasportatore viene attivato tramite "nastroON". Una volta che il pallet arriva in una stazione (attivazione delle fotocellule STAZ_#) il pallet viene sollevato automaticamente dal nastrino (non bisogna gestirlo), così che il nastro possa scorrere per gli altri pallet.
Il sistema dovrà essere in grado di gestire le stazioni in parallelo.

In particolare le stazioni sono:
1.Alimentazione: la stazione attiva "alimentazione" e il pezzo viene posizionato sul pallet. Quando una fotocellula "presenza_pezzo" rileva il pezzo si passa alla stazione successiva.
2.Saldatura: la stazione attiva "saldaturaON" per 5 secondi e si passa alla stazione successiva.
3.Foratura: la stazione attiva "foraturaON" per 3 secondi e si passa alla stazione successiva.
4.Avvitatura: la stazione attiva "avvitaturaON" per 4 secondi e si passa alla stazione successiva.
5.Controllo visivo: la stazione attiva "quality_check". Quando "quality_check_done" è a true il nastro procede scaricando il pezzo in un contenitore (non gestito).

La linea viene disabilitata dopo 10 pezzi, poiché il sistema necessita di manutenzione. In questo caso, viene attivato un allarme MAI e resettato solamente dopo che un operatore preme il tasto di reset manutenzione MAIR.

Gestire le 5 stazioni con un’unica function block.

Esercizio 4 – ST (BONUS EXE)

Istruzioni consegna progetti

12 ore di laboratorio:

- 4 su ladder → progetto su ladder
- 4 su sfc → progetto su sfc
- 4 su st → progetto su st

I progetti DEVONO essere fatti in gruppo (minimo 2 persone, massimo 3)

I 3 progetti verranno valutati complessivamente per massimo 3 punti

I punti dei progetti verranno sommati al voto dello scritto

Inviare i 3 progetti entro il 31/05/2024

Cartella compressa su una cartella condivisa contenente:
- file .txt → nome, cognome e matricola dei componenti del gruppo
- 3 cartelle progetto
- Report pdf dei 3 progetti