

Introduzione all'automazione industriale

Corso Base Machine Controller

OMRON

OEE-I

Di cosa si tratta?

«L'automazione industriale è una disciplina che studia metodi e tecnologie per il controllo di flussi di energia, di materiali e informazioni necessari alla realizzazione dei processi produttivi.»

#Flessibilità

#Qualità

#Energia

#Costo

#Tempo

Il sistema di controllo

«Un sistema di controllo di un processo industriale può essere descritto come un insieme di dispositivi interconnessi e comunicanti tra loro attraverso una o più reti di comunicazione»

«Un dispositivo è caratterizzato da componenti hardware o software che permettono di elaborare i dati e comunicare con l'esterno»



Il sistema di controllo



Omron Forpheus gioca a Ping Pong

Il sistema di controllo



Esempio di soluzione Pick and Place

I prodotti per l'automazione industriale



PLC o MAC

«PLC: Programmable Logic Controller »

«MAC: Machine Automation Controller»

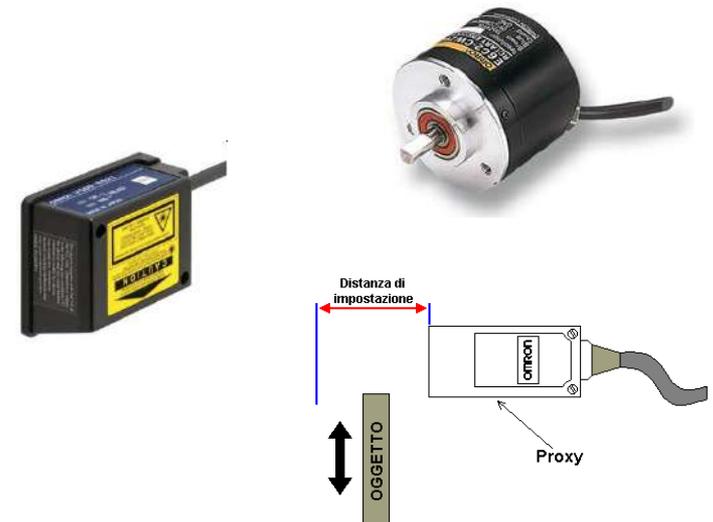
- Acquisisce i segnali dei sensori
- Elabora il programma contenuto nella sua memoria
- Fornisce i segnali di comando agli attuatori



Sensori

Convertono una grandezza fisica in un segnale elettrico

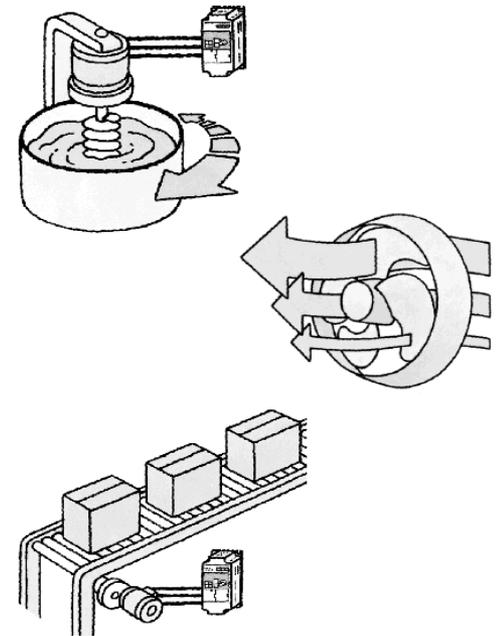
- Movimento, rotazione (encoder)
- Distanza
- Livello
- Temperatura
- Confronto immagini (visione)
- Identificazione
- Presenza (fotocellula, induttivo, capacitivo)



Attuatori

Convertono un segnale elettrico in una grandezza fisica

- Inverter per il controllo della velocità dei motori
- Servo azionamenti (per il controllo della posizione e della coppia)
- Elementi di riscaldamento
- Comandi ON/OFF



Interfacce operatore

Utilizzate dall'operatore per comandare e monitorare lo stato della macchina

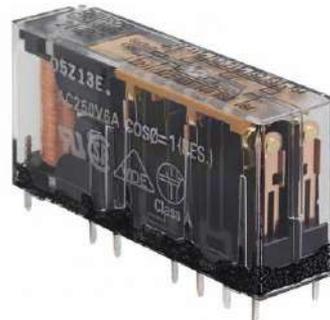
- HMI: Human Machine Interface (pannello operatore)
- IPC: Industrial PC (computer industriale)
- SCADA: Supervisory Control And Data Acquisition (software di supervisione)



Componenti di sicurezza

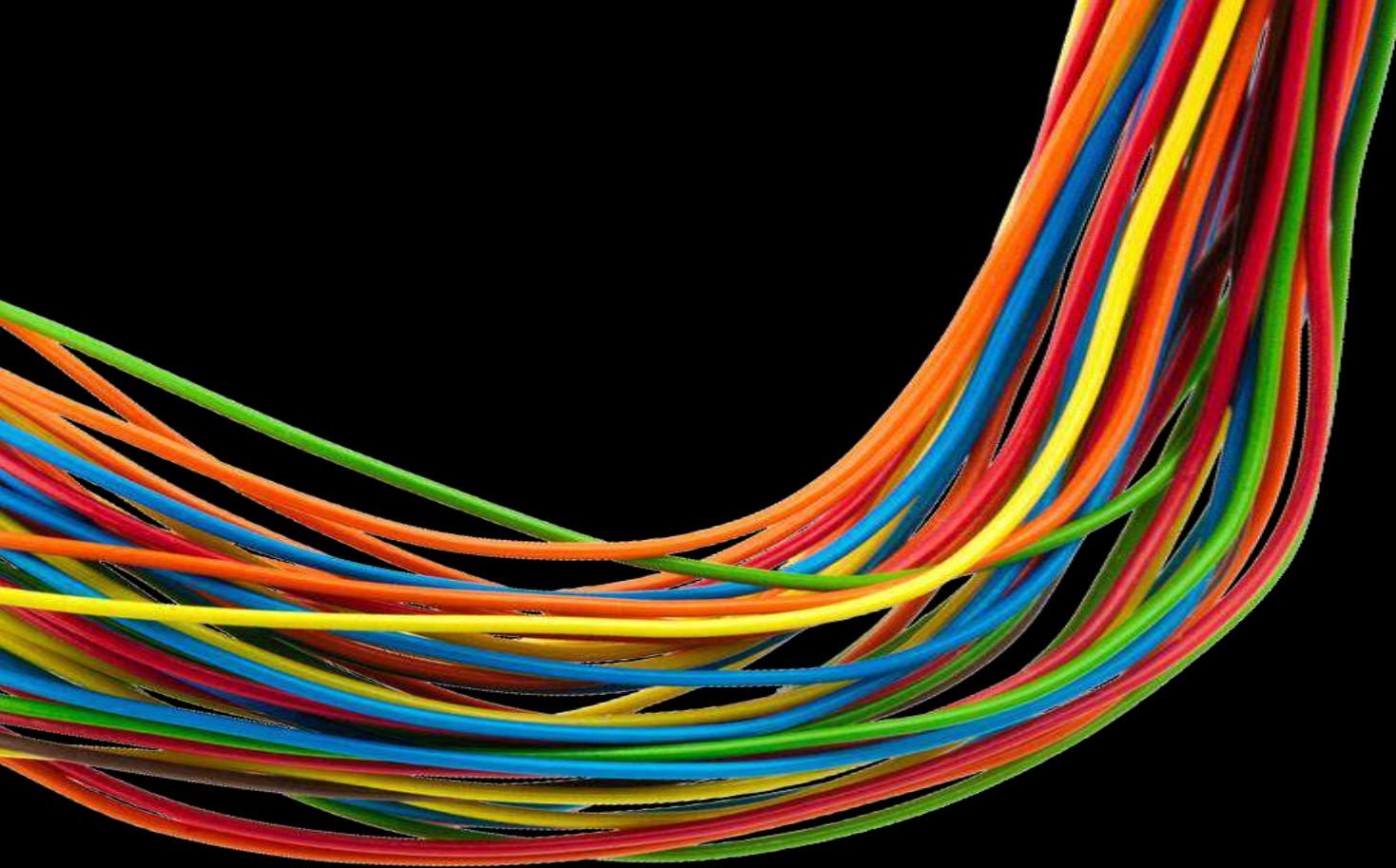
Dispositivi certificati per il funzionamento in sicurezza della macchina

- Controllore di sicurezza
- Sensori di sicurezza
- Relè di sicurezza





DOMANDE?



I segnali

Corso Base Machine Controller

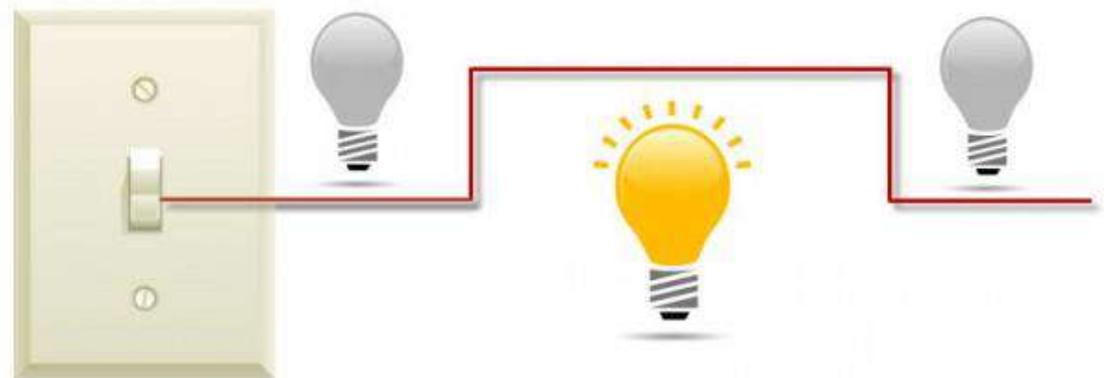
OMRON

OEE-1

Digitali Vs Analogici

I segnali elettrici possono essere:

Digitali



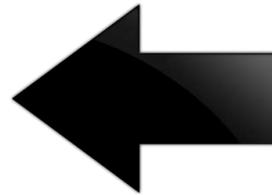
Analogici



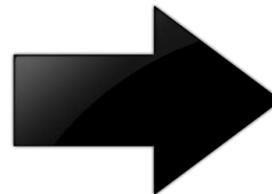
Ingressi Vs Uscite

I segnali elettrici possono essere collegati a:

Ingressi

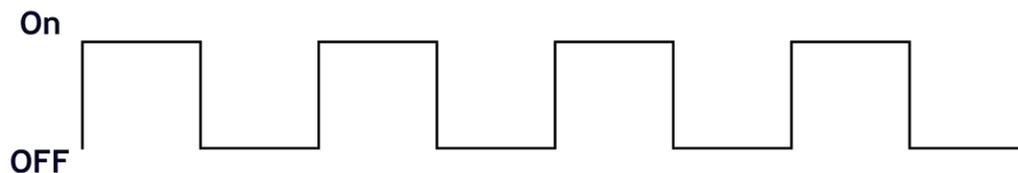


Uscite



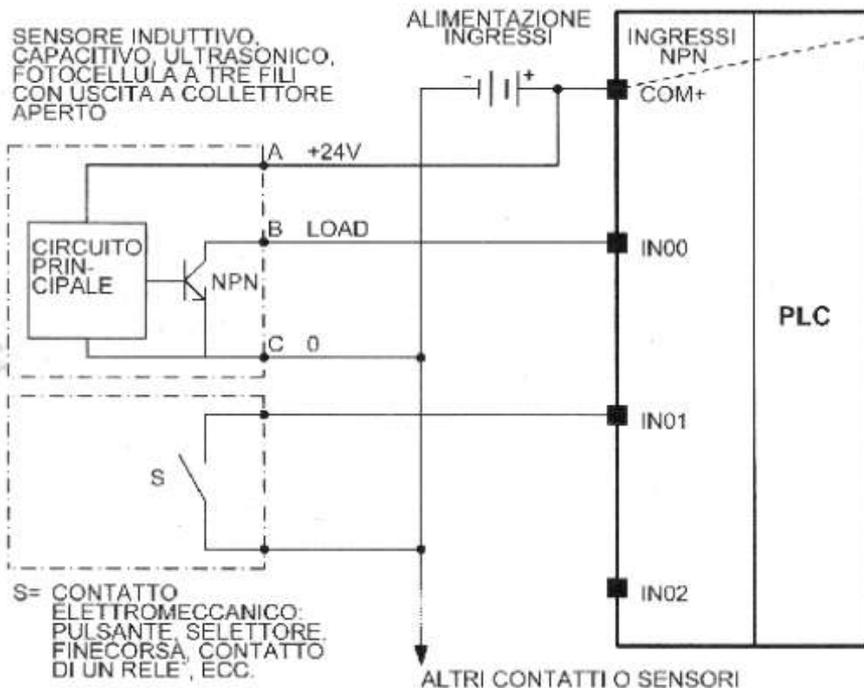
Ingressi digitali

- Il segnale digitale fornisce un'informazione di tipo ON/OFF e viene interpretato dal PLC come la modifica di un bit
- Il livello fisico del segnale digitale è un segnale in tensione
- L'interfaccia fisica di ingresso del PLC deve essere in grado di rilevare il segnale
- Normalmente i segnali digitali rilevabili sono segnali in tensione a 12 o 24 Vc.c.
- È necessario avere informazione sulla logica del segnale ovvero se il segnale è di tipo PNP o di tipo NPN

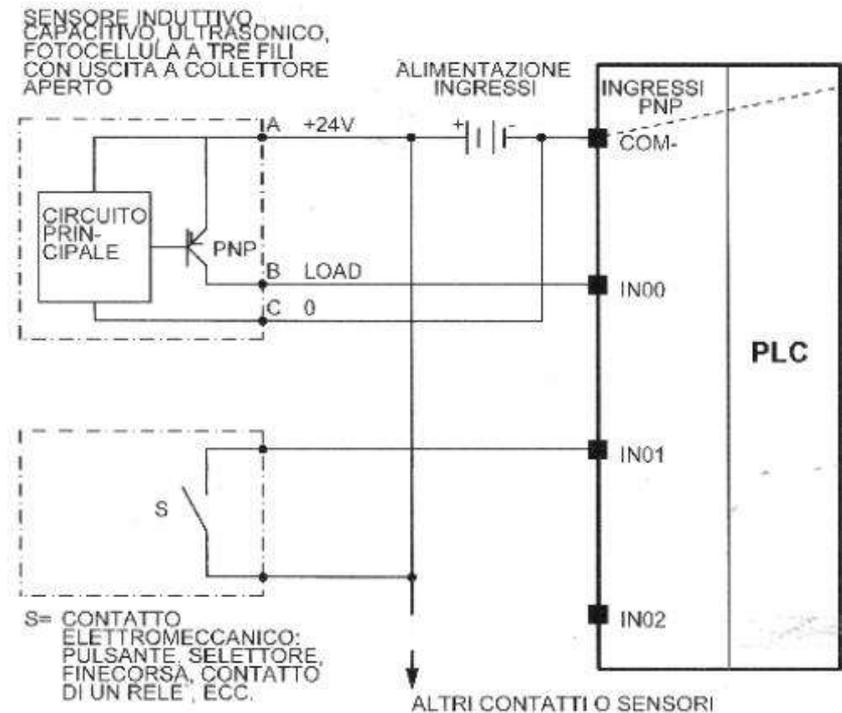


Ingressi digitali: NPN e PNP

- Ingressi NPN



- Ingressi PNP



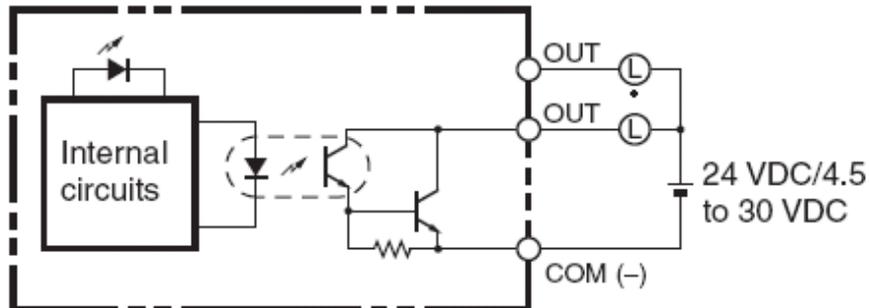
Uscite digitali a transistor

- Il PLC comanda un transistor di uscita. L'entità della corrente sopportabile è dell'ordine delle centinaia di mA
- Il transistor può sopportare solo tensioni continue polarizzate
- La scelta del tipo di uscita a transistor (NPN o PNP) avviene in base al tipo di dispositivo da comandare
- Non ci sono limiti fisici al numero di commutazioni
- La frequenza di commutazione è dell'ordine di grandezza dei kHz (superiore rispetto all'uscita a relè)

Uscite a transistor: NPN e PNP

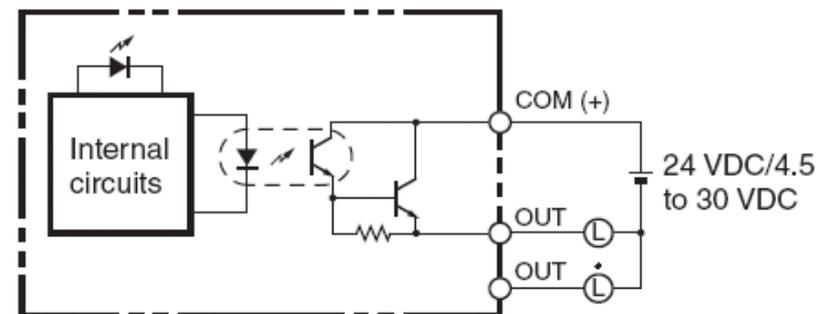
- NPN**

Quando l'uscita si attiva, il transistor conduce: l'OUT e COM vengono portati allo stesso potenziale (**0V**)



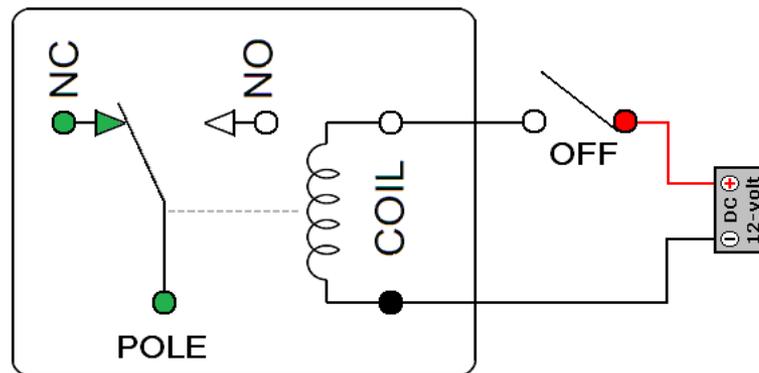
- PNP**

Quando l'uscita si attiva, il transistor conduce: l'OUT e COM vengono portati allo stesso potenziale (**24V**)



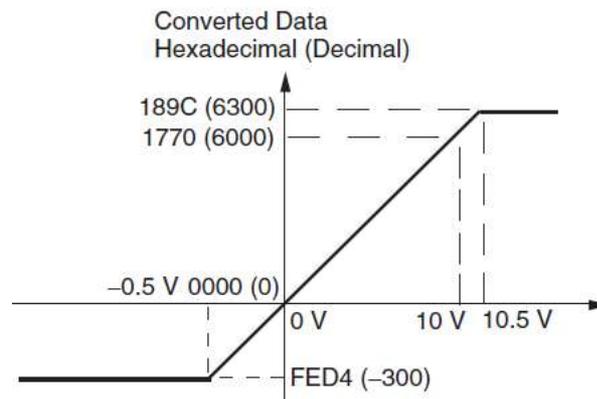
Uscita a relè

- Il PLC comanda un contatto di uscita che può sopportare correnti di una certa entità (qualche Ampere) e segnali in tensione sia continua sia alternata
- Un'uscita a relè ha un numero massimo di operazioni possibili dovuto allo spostamento fisico (frequenza di commutazione) e all'usura del contatto



Segnale analogico

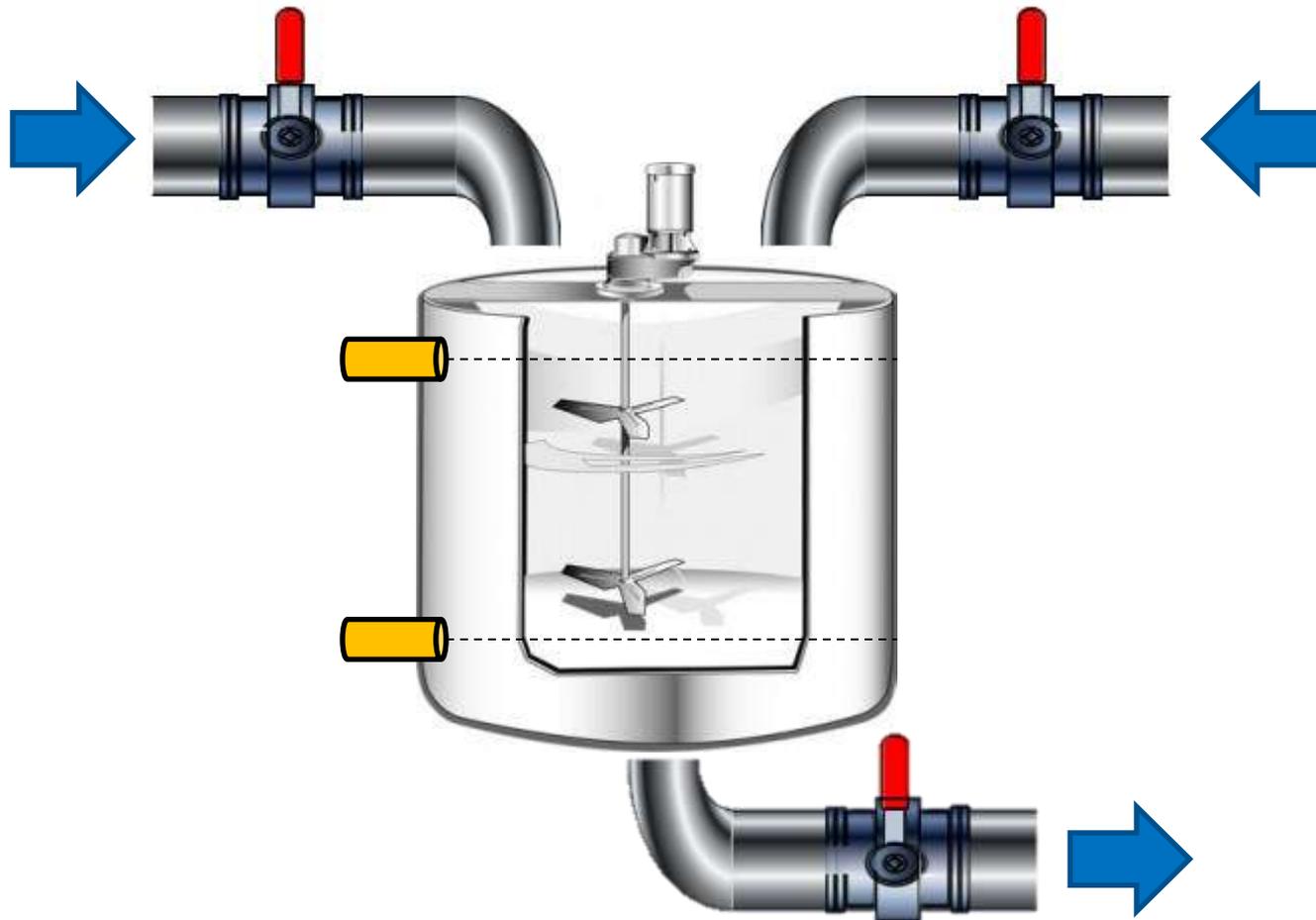
- Il segnale analogico fornisce un'informazione di *tipo complesso*
- Il livello in ingresso, viene convertito in un valore digitale in funzione della risoluzione della scheda di acquisizione e copiato in un'area di memoria del PLC
- Allo stesso modo un valore digitale contenuto in un'area di memoria del PLC viene convertito in un valore di tensione di uscita
- Di seguito è riportato un esempio di conversione su un segnale di ingresso 0-10V



Segnale analogico: Principali caratteristiche

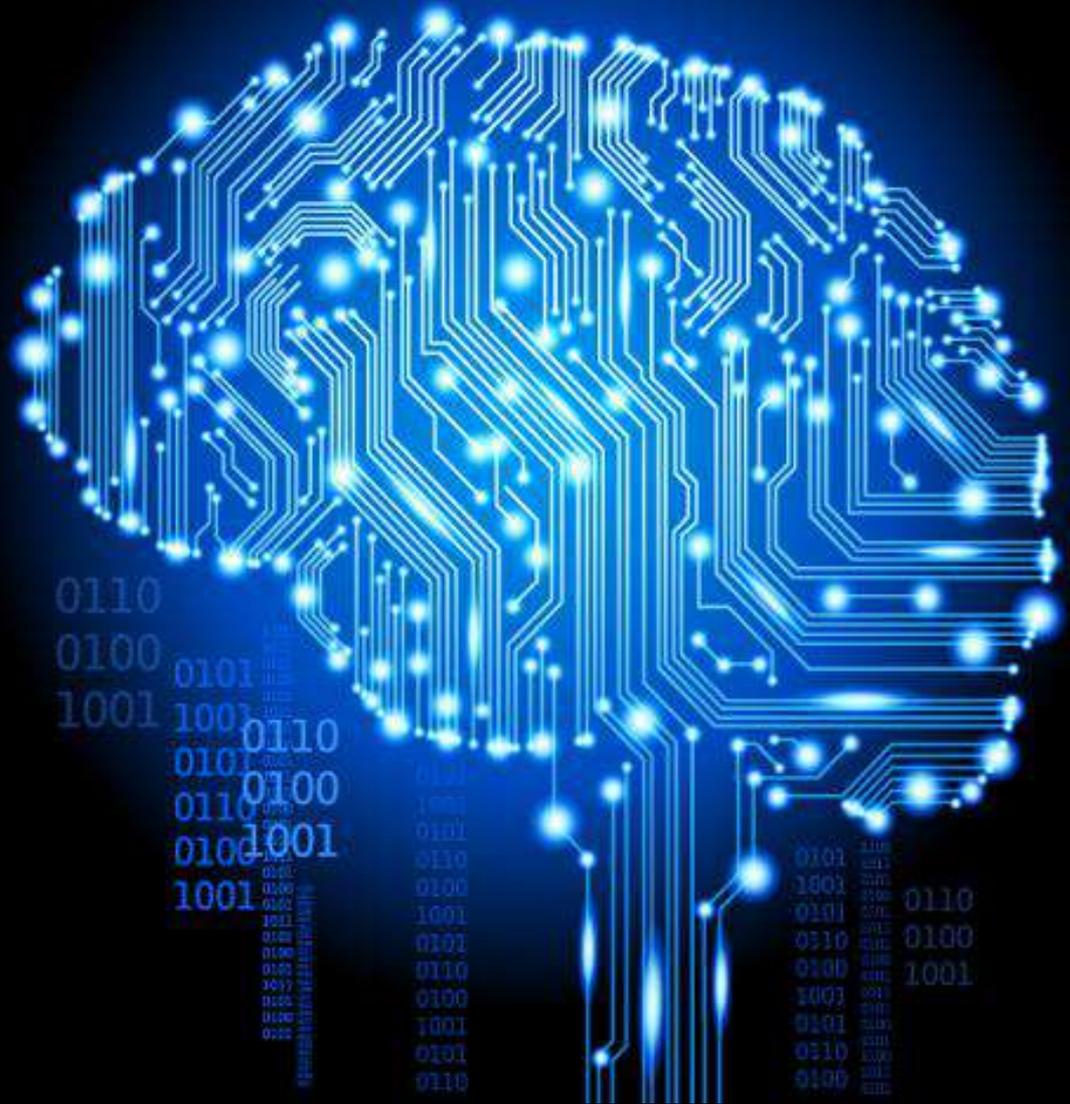
- Risoluzione:
 - Indica la sensibilità nel rilevare variazioni del segnale analogico
- Tempo di campionamento:
 - indica la capacità di inseguire una variazione del segnale di ingresso.
- Tipi di segnali:
 - segnali in tensione 0..10V, -10..+10V, 0..5V o 1..5V
 - segnali in corrente 0..20 mA o 4..20 mA
- Precisione:
 - Percentuale massima di errore sul fondo scala e linearità

Esempio: Mixer





DOMANDE?



MAC: Struttura e logica di funzionamento

Corso Base Machine Controller

OMRON

OEE-I

PLC: definizione

- Programmable Logic Controller: Controllore a Logica Programmabile

« APPARECCHIATURA ELETTRONICA PROGRAMMABILE PER IL CONTROLLO DI MACCHINE E PROCESSI INDUSTRIALI »

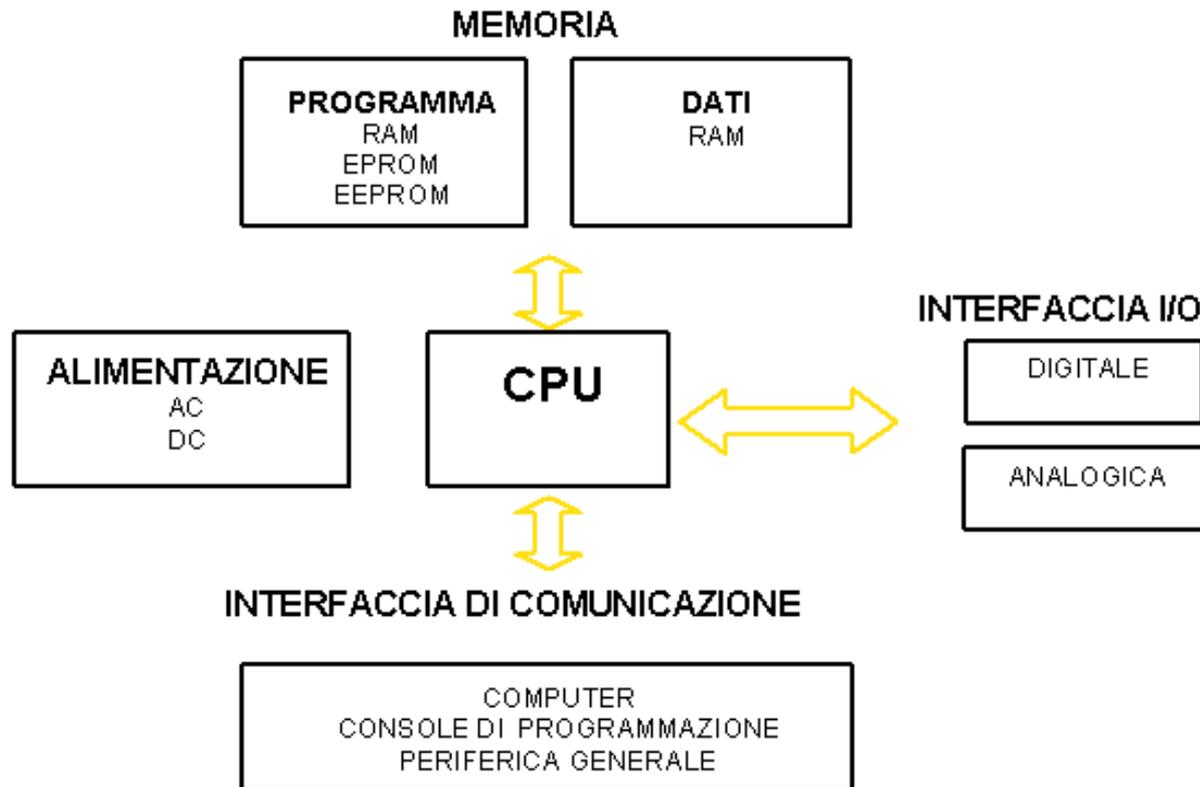
- Nasce come elemento sostitutivo della logica cablata e dei quadri di controllo a relè
- Si qualifica in breve tempo come elemento insostituibile nell'automazione di fabbrica

La logica PROGRAMMATA

- La logica di controllo è realizzata tramite una serie di ISTRUZIONI caricate nella Memoria di Programma
 - E' facile da ampliare e modificare
 - E' di facile diagnostica
 - E' facilmente interfacciabile con dispositivi quali PC, stampanti, terminali,
 - E' pensata per il controllo di sistemi complessi

PLC: la struttura

- In un PLC possiamo distinguere i seguenti blocchi funzionali



La CPU

- E' il "cervello" del sistema dove avvengono le elaborazioni delle informazioni e la gestione del PLC stesso
- Le sue caratteristiche sono:
 - la velocità di elaborazione
 - la memoria di programma
 - la memoria dati
 - il numero e il tipo di funzioni integrate disponibili
 - il numero di interfacce gestibili
 - I/O, Comunicazione e Memoria

La memoria

Nei PLC Sysmac si divide in memoria dati e memoria programma

- La memoria **Programma** contiene la sequenza delle istruzioni di programma. È normalmente una memoria ritentiva. Tutte le CPU Sysmac dispongono di una memoria di backup su scheda SD.
- La memoria **Dati** contiene i dati da elaborare: la memoria dati viene utilizzata per allocare le variabili del programma.

La memoria variabili

- La memoria per le variabili è suddivisa in due parti:
 - Area per le variabili RITENTIVE: sono quelle variabili che permettono di mantenere i dati salvati anche dopo lo spegnimento dell'unità o al cambio della modalità operativa
 - Area per le variabili NON RITENTIVE: sono le variabili che si azzerano o tornano al valore iniziale dopo lo spegnimento dell'unità o al cambio della modalità operativa

Le interfacce e l'alimentatore

- **Interfaccia di I/O** (analogica o digitale): è l'hardware dedicato al collegamento di sensori e attuatori. Questa interfaccia permette di convertire il segnale fisico (elettrico) in un segnale logico (bit o word) dell'area di memoria
- **Interfaccia di comunicazione**: permette la comunicazione del PLC con altri dispositivi che hanno una porta seriale (PC, PLC, Inverter, MMI, lettori di codici a barre, termoregolatori, RFID...)
- **Alimentatore**: fornisce l'alimentazione a tutto il sistema PLC. Deve essere dimensionato in funzione della potenza assorbita dai moduli collegati

DAL PLC AL MAC

M1R



CQM1



CP1



NX1P



C20



CJ1



NJ/NX

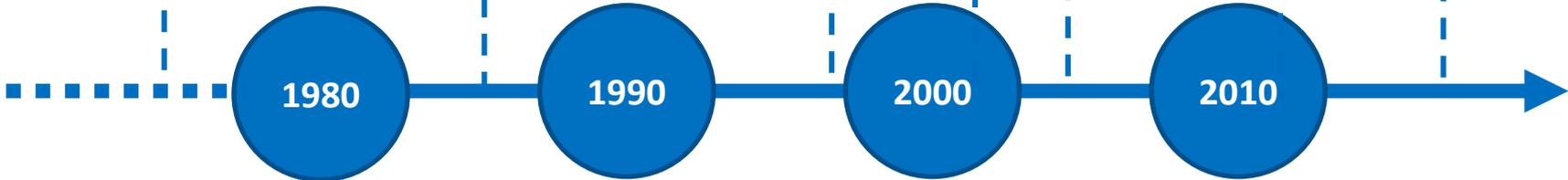


1980

1990

2000

2010



Sysmac: PLC + MC = MAC

E' la più recente serie di PLC Omron destinata all'automatizzazione di un'intera macchina in tutti i suoi aspetti.

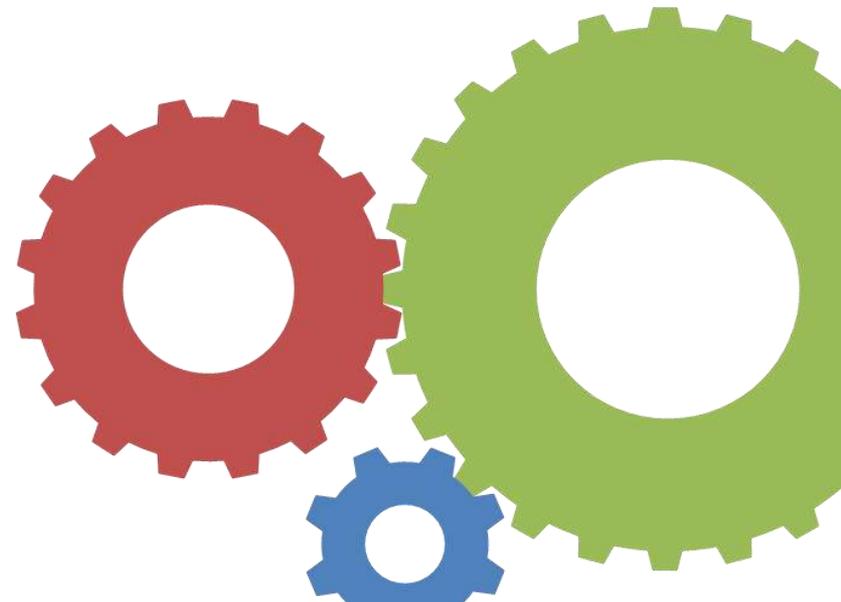
MAC = Machine Automation Controller

La piattaforma Sysmac è stata sviluppata al 100% da Omron, seguendo delle specifiche provenienti dai costruttori di macchine, secondo criteri innovativi.



MAC: integrazione Logic / Motion

- Due «motori» Logic e Motion lavorano in completa sincronia all'interno del ciclo di controllo della CPU;
- Le funzioni motion sono eseguite direttamente dal «motore» Motion;
- Tutte le funzionalità del motion controller sono accessibili dal programma PLC sfruttando Blocchi Funzione aderenti allo standard PLC Open



MAC compatti e modulari

I MAC compatti sono composti da un unico “blocco” contenente alimentazione, CPU e I/O

Anche se è possibile espandere il MAC alle esigenze del sistema da controllare, l’espandibilità complessiva è inferiore a quella di un MAC modulare.



MAC compatti e modulari

I MAC modulari sono composti da una serie di moduli collegati tra loro tramite bus di comunicazione interno

Alimentatore, CPU e moduli di I/O sono quindi unità indipendenti e ordinabili separatamente

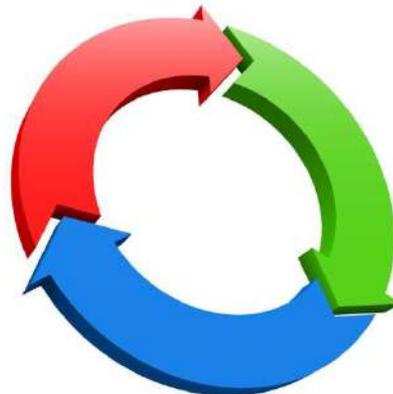


Il MAC è liberamente configurabile in base alle necessità del sistema da controllare

MAC: logica di funzionamento

L'elaborazione dei dati di un MAC avviene di norma in modo sequenziale secondo l'ordine delle istruzioni all'interno del programma

La differenza con un normale computer dipende dal fatto che l'attuazione (segnale fisico sull'uscita) e l'acquisizione (rilevamento dell'ingresso) avvengono di norma alla fine dell'elaborazione (a meno di specifiche funzioni)



MAC: logica di funzionamento

- Possiamo considerare la logica di controllo di un MAC suddivisa in tre momenti:
 1. Acquisizione dei dati dai sensori (fotografia degli ingressi)
 2. Elaborazione dei dati + processi comuni
 3. Attuazione delle uscite



MAC: logica di funzionamento

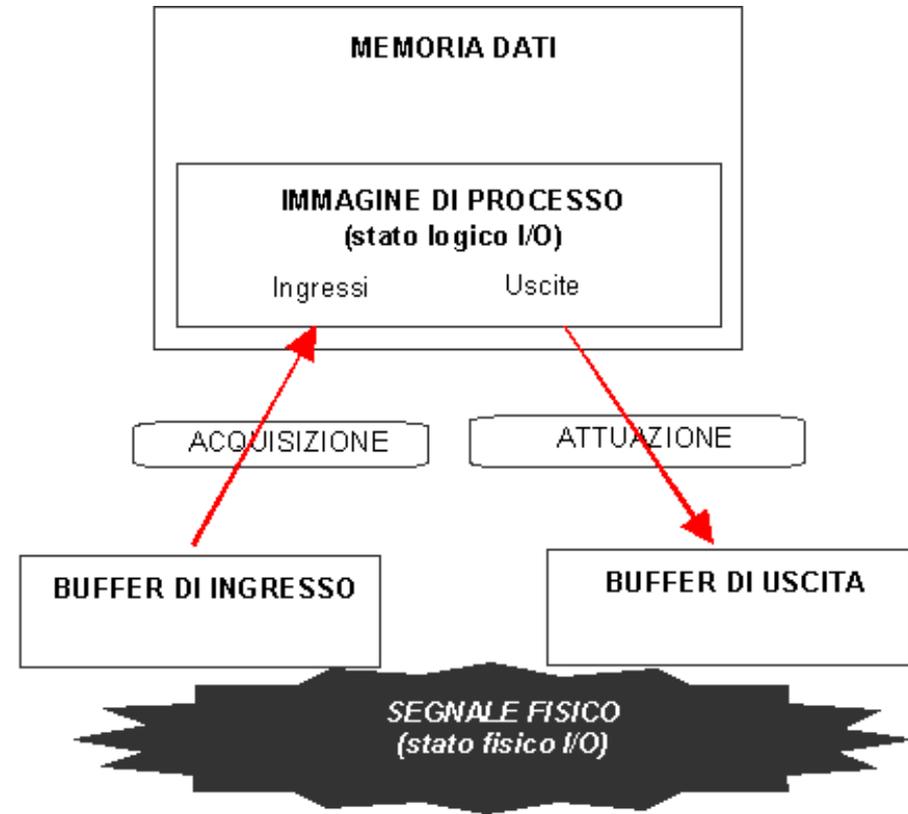
Questo modo di funzionamento implica l'impossibilità di rilevare (se non con speciali funzioni o moduli) la variazione di un segnale che avviene durante l'elaborazione del programma

- **Nella prima fase** viene acquisita in un preciso istante la condizione degli ingressi (fotografia degli ingressi fisici) e fatta una "copia" nell'area di memoria di I/O
- **Nella seconda fase** vengono elaborati gli ingressi e modificate le uscite (solo all'interno dell'area di memoria del MAC) in modo sequenziale secondo l'esecuzione del programma
- **Nella terza fase** avviene l'attuazione fisica delle uscite secondo lo stato logico determinato durante l'elaborazione del programma

La prima e la terza fase vengono comunemente riunite in un'unica fase di "I/O Refresh" (rinfresco di Ingressi e Uscite)

PLC e l'Immagine di Processo

- L'Immagine di Processo contiene lo stato logico degli ingressi relativo all'ultima acquisizione e lo stato logico delle uscite risultante dall'elaborazione del programma
- Le uscite verranno fisicamente aggiornate al corrispondente stato logico solo all'attuazione successiva



La modalità operativa del MAC

Il MAC ha 2 modalità operative:

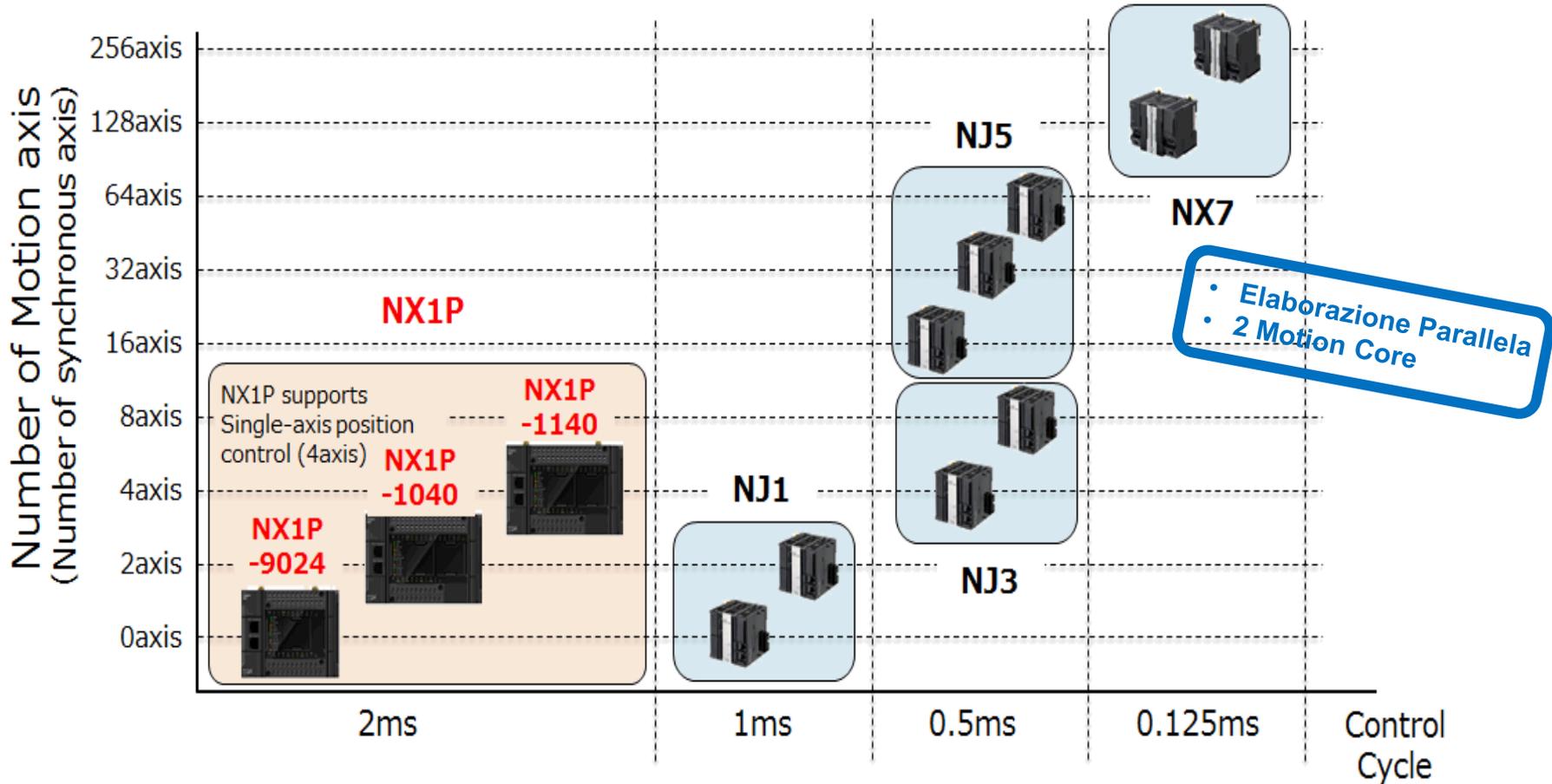
- Program: il programma non viene eseguito; è possibile scaricare il programma e le impostazioni del MAC
- RUN: il programma viene eseguito in modo ciclico; è possibile scrivere forzare le variabili di ingresso e uscita ma non è possibile scaricare il programma e le impostazioni del MAC (modalità tipica di funzionamento)



Controllore NX1P2

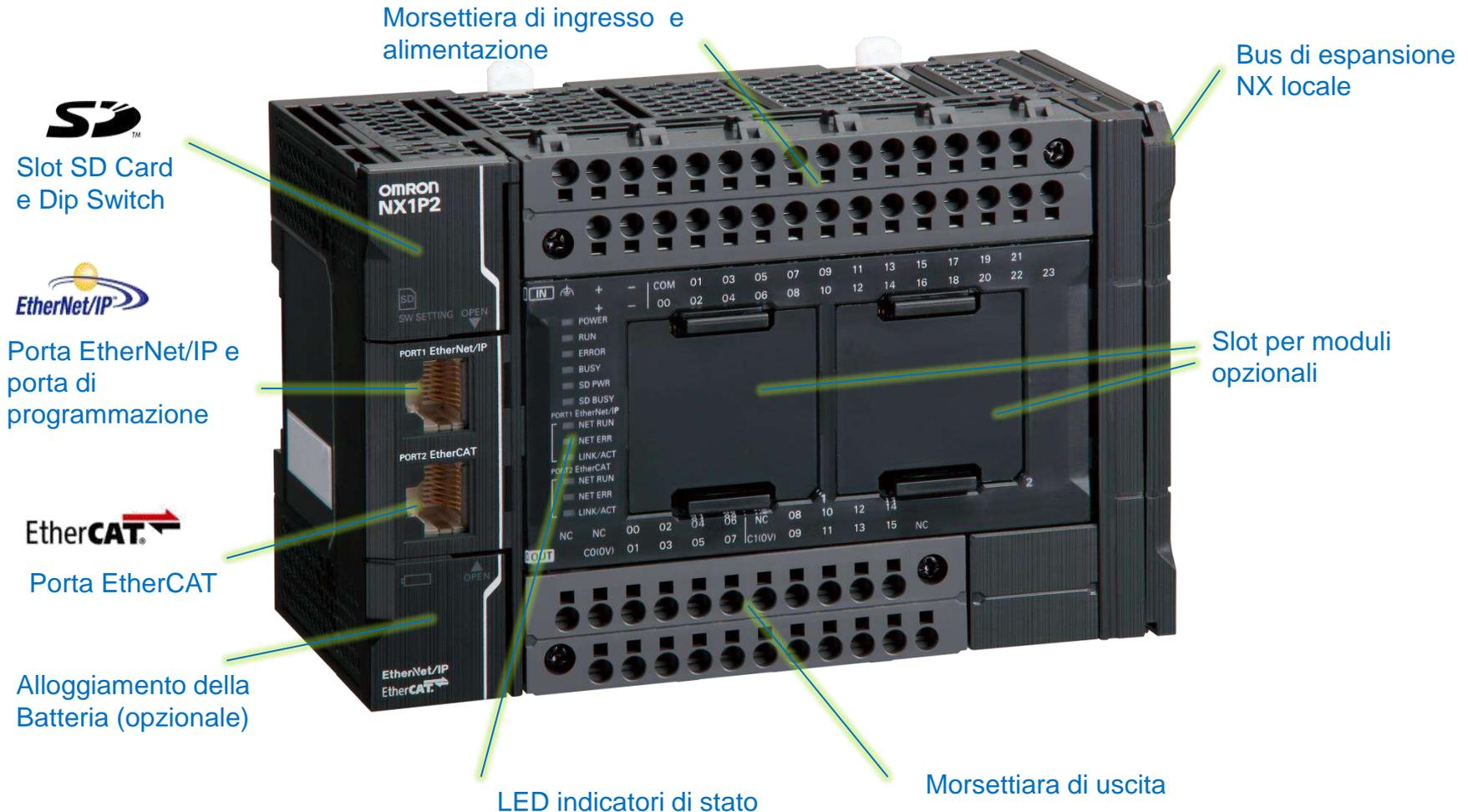
Corso Base Machine Controller

Famiglia di Controllori Sysmac



La CPU NX1P2

Aspetto esteriore



Gamma prodotti

La famiglia NX1P2 è composta da 6 modelli, 3 con uscite a transistor PNP (finale 1) e 3 con uscite a transistor NPN. L'alimentazione è sempre 24V DC.

Model	I/O integrati	Uscite	Numero Slot	Assi totali
NX1P2-1140DT1	40 (24 In, 16 Out)	PNP	2	12
NX1P2-1040DT1	40 (24 In, 16 Out)	PNP	2	10
NX1P2-9024DT1	24 (14 In, 10 Out)	PNP	1	4
NX1P2-1140DT	40 (24 In, 16 Out)	NPN	2	12
NX1P2-1040DT	40 (24 In, 16 Out)	NPN	2	10
NX1P2-9024DT	24 (14 In, 10 Out)	NPN	1	4

Classificazione CPU – PLC compatti

Le caratteristiche principali che permettono di classificare una CPU sono le seguenti:

- Memoria programma
- Memoria dati
- Numero di I/O massimi gestibili
- Numero di moduli di espansione collegabili
- Tempo di esecuzione delle istruzioni
- Linguaggi di programmazione supportati

Aree di memoria variabili

La memoria per le variabili è suddivisa in due parti:

- Area per le variabili RITENTIVE: sono quelle variabili che permettono di mantenere i dati salvati anche dopo lo spegnimento dell'unità o al cambio della modalità operativa
- Area per le variabili NON RITENTIVE: sono le variabili che si azzerano o tornano al valore iniziale dopo lo spegnimento dell'unità o al cambio della modalità operativa

Batteria

- Le CPU NX1P2 dispongono di un alloggiamento per la batteria (opzionale)
- Poiché oltre al programma, l'NX1P2 salva anche le Variabili Ritentive sulla propria memoria Flash integrata, a PLC spento la batteria è esclusivamente necessaria al mantenimento dell'orologio calendario
- In assenza di alimentazione, l'NX1P è indicativamente in grado di mantenere l'orologio calendario per circa 10 giorni a 25°C, grazie alla capacità dei condensatori

Specifiche comuni a tutte le CPU NX1P2

Memoria programma	Dimensione	1.5 Mbyte
	Definizioni POU	450
	Istanze POU	1800
Memoria variabili ritentive	Dimensione	32kB
	Numero variabili	5000
Memoria variabili non ritentive	Dimensione	2 MB
	Numero variabili	90000

Nota: 32kB = 32000 byte = 16000 word

Espandibilità e configurazione

La configurazione è espandibile con l'aggiunta di moduli I/O serie NX sia localmente che su rete EtherCAT.

Esistono moduli di espansione:

- Ingressi/uscite relè/transistor
- Ingressi/uscite analogici
- Ingressi per termocoppie o PT100
- Ingressi per Encoder
- Uscite a treno di impulsi
- Unità speciali: ingresso celle di carico, master IO-Link, ingresso Heater Burnout.

Espandibilità locale con I/O NX

- Tutte le CPU NX1P2 accettano fino a **8** moduli di espansione collegati al BUS locale



- **NOTA BENE:** Alcuni moduli NX I/O necessitano di un modulo di alimentazione ausiliaria (NX-PF) sul BUS locale, che viene conteggiato come un modulo!

Schede opzionali – Gamma

NX1W-CIF01: porta RS-232C. Max distanza di trasmissione 15 m



NX1W-CIF11: porta RS-422A/485. Max dist. di trasmissione 50 m



NX1W-CIF12: porta RS-422A/485 (isolato). Max dist. di trasmissione 500 m



NX1W-ADB21: 2 ingressi analogici, 0-10 V, 0-20 mA



NX1W-DAB21V: 2 uscite analogiche, 0-10 V



NX1W-MAB221: 2 ingressi analogici 0-10 V, 0-20 mA e 2 uscite 0-10 V



Espandibilità via EtherCAT

Le CPU NX1P2 possono controllare fino a un massimo di 16 dispositivi EtherCAT

E' possibile utilizzare un massimo di 16 moduli I/O NX nella rete EtherCAT



Sommario Famiglia SYSMAC:

Modello:		NX1P:	NJ101:	NJ301:	NJ501:	NX701:
Ciclo di Controllo		Da 2 a 8ms Incremento: 250us	1/2/4ms	0.5/1/2/4ms	0.5/1/2/4ms	0.125/0.25 to 8ms Incremento: 250us
Capacità Programma		1.5MB	3MB	5MB	20MB	80MB
Capacità Variabili	Non Ritentive	2MB	2MB	2MB	4MB	256MB
	Ritentive	32kB	0.5MB	0.5MB	2MB	4MB
Numero di moduli collegabili lato CPU (Slave ECAT non inclusi)		NX-I/O 8units	CJ1W 40units	CJ1W 40units	CJ1W 40units	None
	Numero Max IO (I/O ECAT esclusi)	256pti	2560pti	2560pti	2560pti	--
Motion control	Assi Sincroni Motion	0/2/4	0/2	4/8	16/32/64	128/256
	Assi PTP	4	--	--	--	--
Porta USB port		Nessuna	1 porta	1port	1port	1port
Porta Ethernet/IP		1 porta (100M)	1 porta (100M)	1 porta (100M)	1 porta (100M)	2 porte (1G)
Porta EtherCAT		1 porta	1 porta	1 porta	1 porta	1 porta
	Numero Slave ECAT	16 (Max 16 moduli NX I/O)	64	192	192	512
I/O Integrati		Modello 24pti I/O : IN14/OUT10 Modello 24pti I/O : IN24/OUT16				



Basi di Sysmac Studio

Corso Base Machine Controller

OMRON

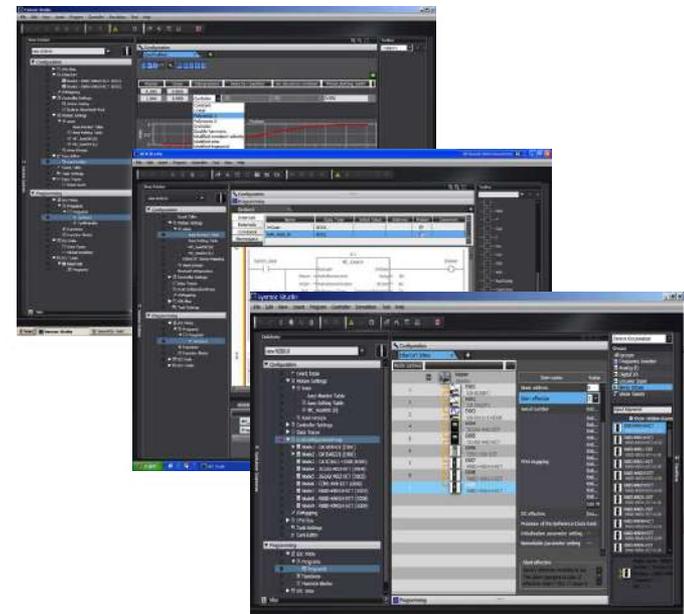
OEE-I

Sysmac Studio

Sysmac Studio è il software sviluppato per integrare le funzioni necessarie alla progettazione delle macchine.

Aspetti principali:

- Programmazione Logica e Motion Control.
- Configurazione I/O e Assi.
- Configurazione Reti: EtherCAT.
- Monitoraggio.
- Simulazione.
- Creazione di profili e camme.



Creazione di un progetto



Doppio clic sull'icona sul desktop.

(Oppure selezionare **Programs – Omron – Sysmac Studio - Sysmac Studio** dal menu Start).

Si hanno tre possibilità:

1. Creare un Nuovo progetto;
2. Richiamare un progetto esistente dalla libreria;
3. Importare un file di progetto sfogliando l'hard disk.



Proprietà del progetto

Cliccando nuovo progetto si aprirà la finestra delle proprietà del progetto.

Nome progetto

Autore (optional)

Commento (optional)

Tipo Progetto

Prima Periferica

Crea

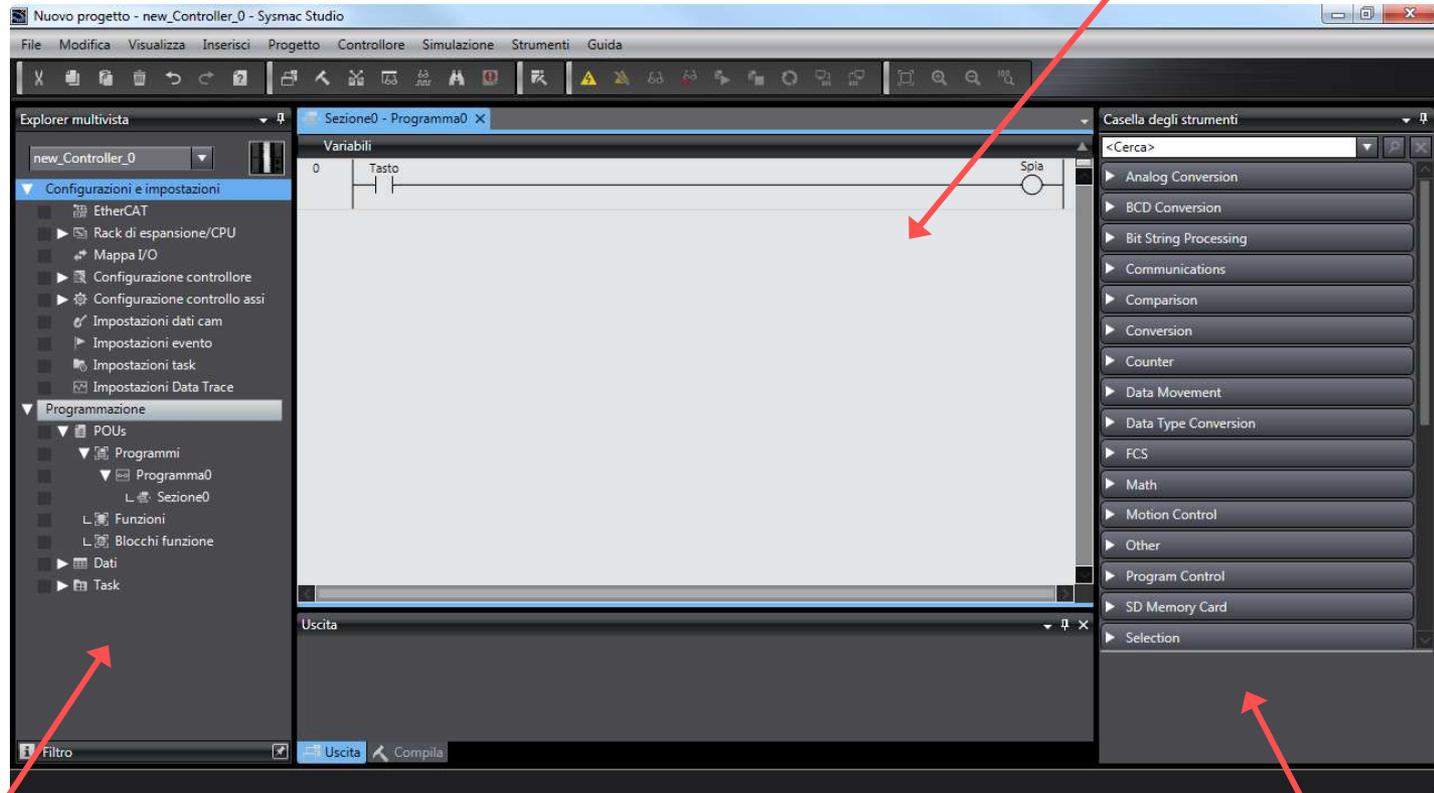
Una volta compilati i campi necessari premere sul pulsante **Crea**

Panoramica

Barra dei menu

Spazio di Lavoro

Barra delle icone



Explorer Multivista

Risultati delle compilazioni, delle ricerche e monitoraggio

Casella degli strumenti

Importare, Esportare, Salvare e Chiudere il progetto

Chiudi: closes the project (will be prompted whether to save the project before closing);

Salva: immagazzina il progetto in libreria;

Salva come...: immagazzina il progetto in libreria con un nuovo nome;

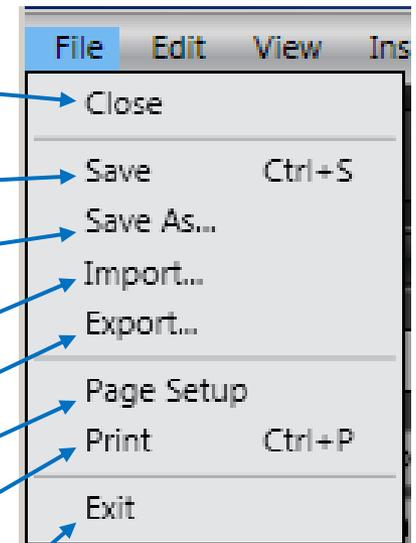
Importa: apre un progetto immagazzinato sul disco rigido;

Esporta: salva il progetto in una cartella del disco rigido;

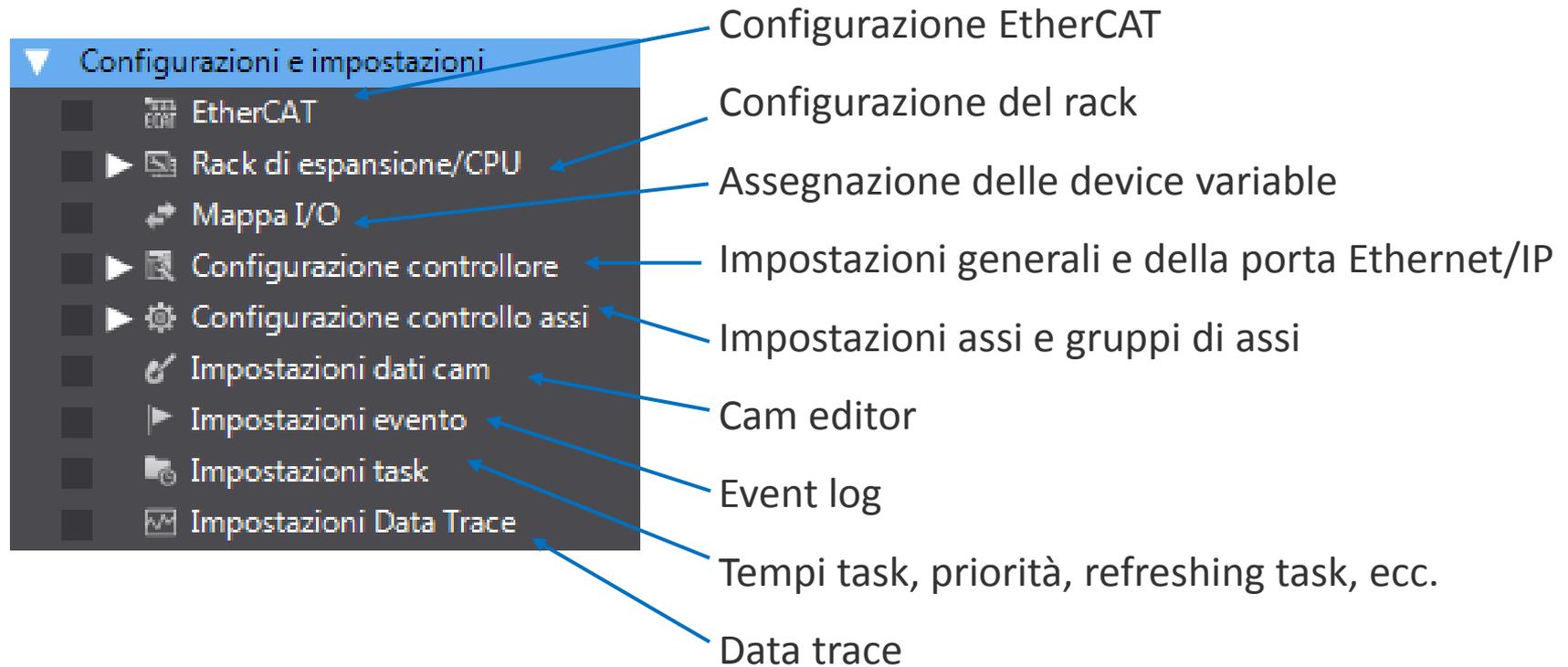
Imposta pagina: imposta margini, intestazione, piè di pagina e carattere per la stampa del progetto

Stampa: Stampa il progetto o parti di esso (come programmi, configurazioni, tabelle variabili...);

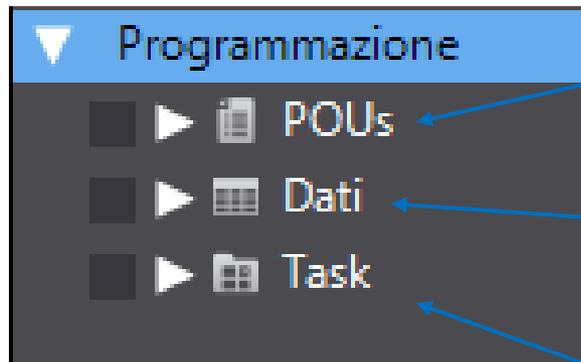
Esci: chiude il progetto ed esce da Sysmac Studio (verrà mostrato un popup per salvare il progetto)



Explorer multivista: Configurazioni



Explorer multivista: Programmazione



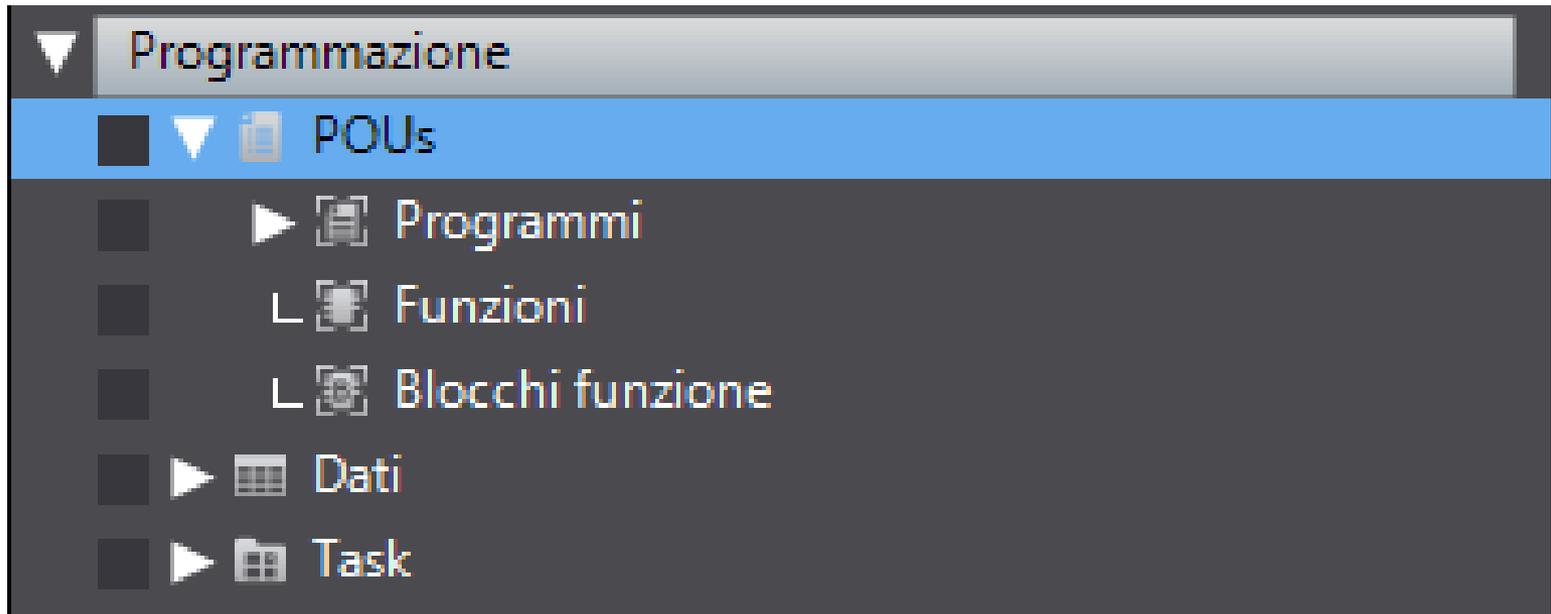
Editor programmi, FB e FUN

Definizione nuovi tipi di dati e variabili globali

Organizzazione task

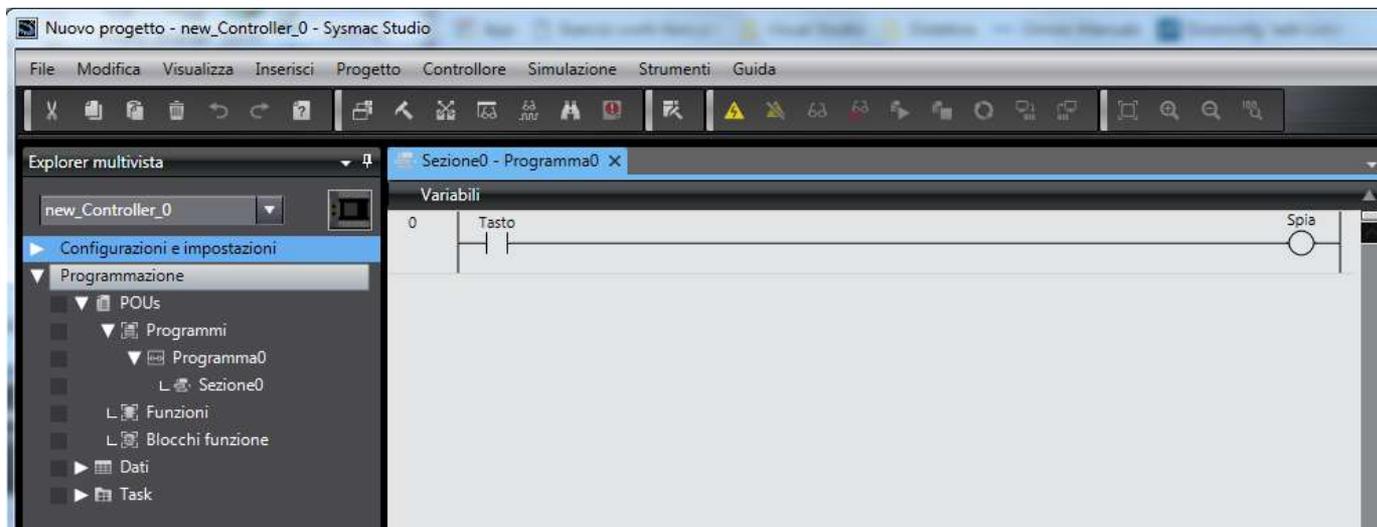
POU (Program Organization Units)

Un POU è un'unità contenente codice da eseguire. Può essere Programma, Funzione o Blocchi funzione.



POU: Editor ladder

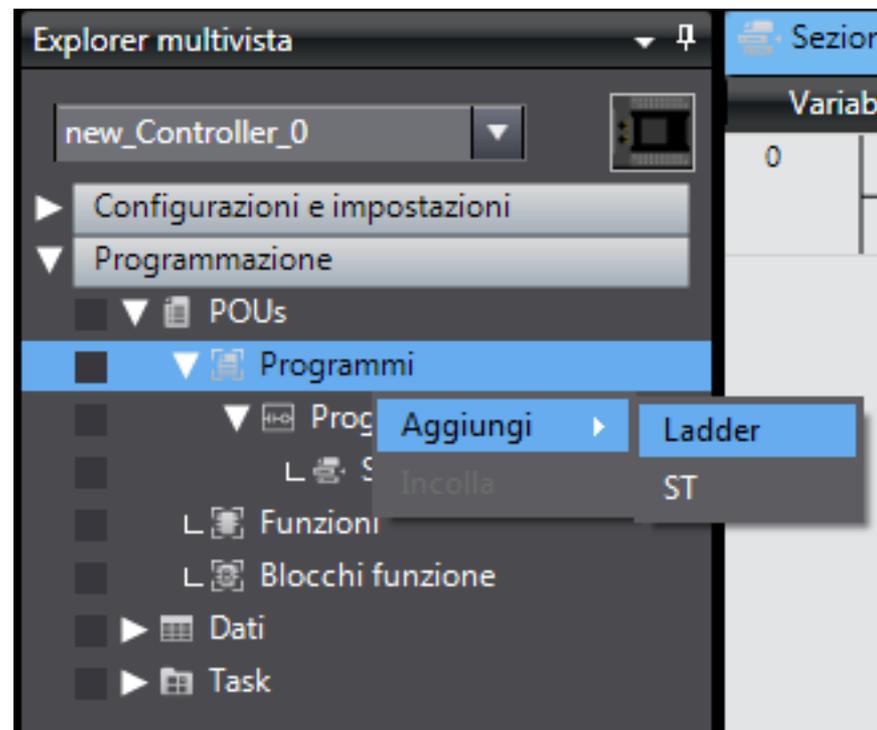
1. Dalla scheda POU espandere la vista Programmi
2. Doppio clic su **Sezione0**
3. Verrà visualizzata la tabella variabili e l'**Editor Ladder**



Nella scheda Interni si dichiarano le variabili locali, in quella esterni si richiamano le variabili globali già definite.

POU: Aggiungere un programma Ladder

1. Selezionare **Programmazione – POU** nell'Explorer multivista e fare clic col destro su Programmi
2. Seleziona **Aggiungi – Ladder**



POU: Editor dei programmi

The screenshot displays the OMRON POU editor interface. At the top, there is a tab labeled 'Sezione0 - Programma0'. Below it, a 'Variabili' section is expanded to show a table of local variables. The table has three columns: 'Nome', 'Tipo dati', and 'Valore inizi'. It lists two internal variables: 'Spia' and 'Tasto', both of type 'BOOL'. A green arrow points from the 'Variabili locali Interne / Esterne' text below to the 'Interne' tab of the table. To the right of the table is a 'Casella degli strumenti' (Toolbox) with a search bar and various function blocks like 'Analog Conversion', 'BCD Conversion', etc. Below the table is the 'Ladder / ST editor' showing a simple ladder logic diagram with a normally open contact labeled 'Tasto' and a coil labeled 'Spia'.

	Nome	Tipo dati	Valore inizi
Interne	Spia	BOOL	
Esterne	Tasto	BOOL	

Variabili locali **Interne** / **Esterne**

Nota: la registrazione di variabili locali segue le stesse regole di quelle globali

Dati: Variabili globali

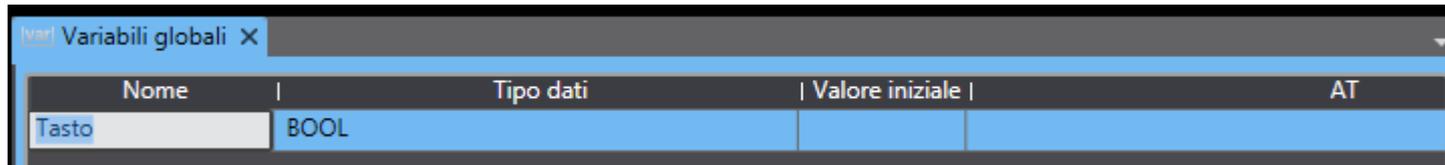
The screenshot displays the 'Variabili globali' (Global Variables) window in the OMRON GX Developer software. The interface is split into two main sections:

- Left Pane (Explorer multivista):** Shows a hierarchical project tree for 'new_Controller_0'. The tree includes 'Configurazioni e impostazioni', 'Programmazione', 'POUs', 'Programmi', 'Programma0', 'Sezione0', 'Funzioni', 'Blocchi funzione', 'Dati', 'Tipi dati', 'Variabili globali' (highlighted), and 'Task'.
- Right Pane (Table):** Displays an empty table with the following structure:

Nome	Tipo dati	Valore iniziale
<i>Vuoto. Per aggiungere un elemento fare clic qui.</i>		

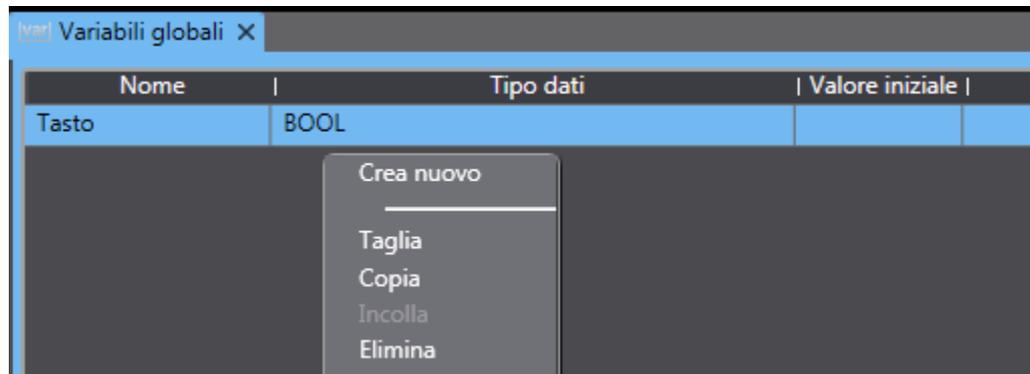
Dati: Variabili globali

1. Posizionare il cursore nella tabella Variabili Globali
2. Impostare tutti i campi e premere invio.



Nome	Tipo dati	Valore iniziale	AT
Tasto	BOOL		AT

3. Per aggiungere nuove variabili premere il tasto della tastiera **Insert**, oppure cliccare col tasto destro e selezionare Crea nuovo dal menu.



Nome	Tipo dati	Valore iniziale	AT
Tasto	BOOL		

Crea nuovo

Taglia

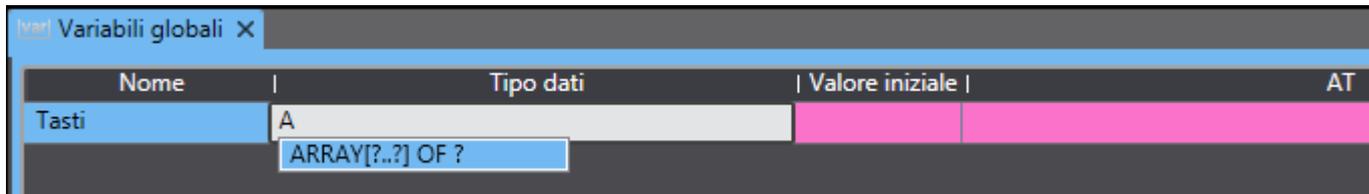
Copia

Incolla

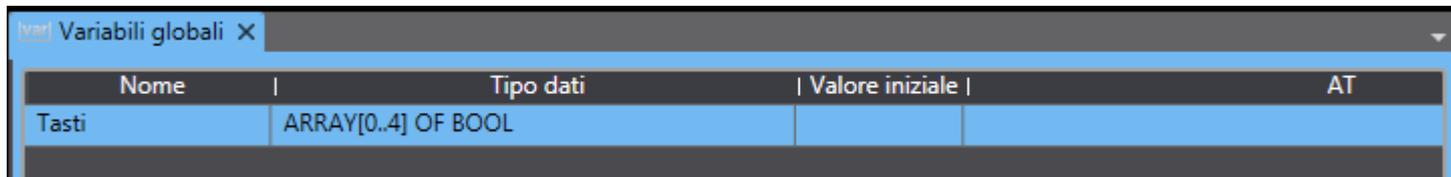
Elimina

Definire Variabili Array

1. Posizionare il cursore nella tabella delle variabili globali;
2. Nel campo Tipi di dati scrivere "A" e quindi selezionare "ARRAY[?..?] OF ?" dal menu di autocompletamento;



3. A questo punto indicare la dimensione dell'array e il data type degli elementi



Note:

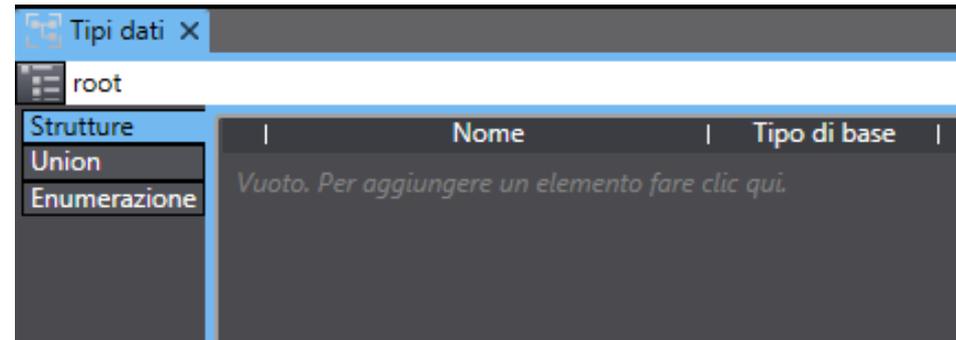
- L'elemento iniziale dell'array non deve essere necessariamente 0
- Gli array possono essere definiti sia per variabili globali che locali

Definire tipi di dati derivati

Si possono definire tipi di dati derivati combinando quelli esistenti

Per aprire l'editor dei tipi di dati:

1. Espandere il menu Dati;
2. Doppio clic su Tipi di dati.



I tipi di dati derivati definibili sono:

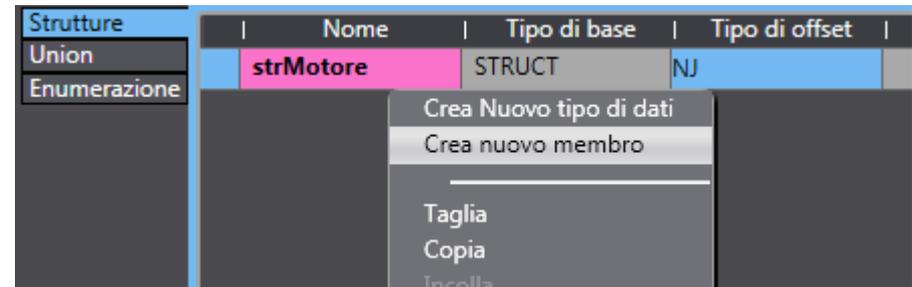
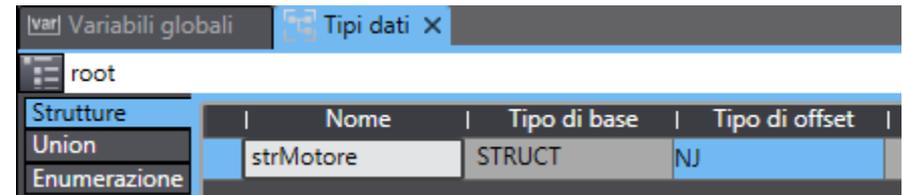
- **Strutture**
- **Unioni**
- **Enumerazioni.**

Definire tipi di dati derivati: Strutture

Le strutture sono tipi di dati “multielement” che raggruppano dati di tipi diversi in un’unica variabile.

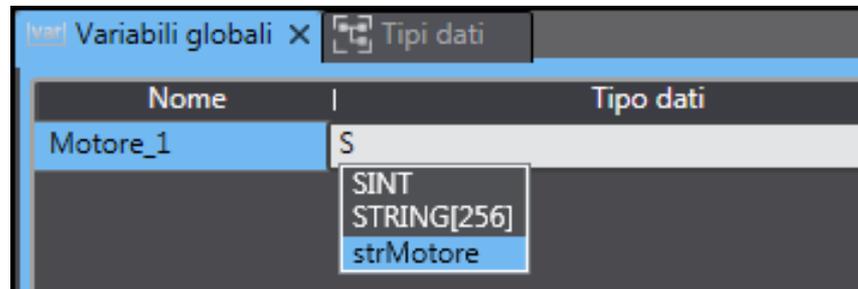
Dall’editor dei tipi di dati:

1. Selezionare la scheda **Strutture**;
2. Cliccare sull’editor
3. Compilare il campo Nome e premere invio;
4. Cliccare con il tasto destro del mouse e selezionare aggiungi membro;
5. Scrivere il nome e il tipo di dati del membro;
6. Stessa procedura per aggiungere ulteriori membri alla struttura.



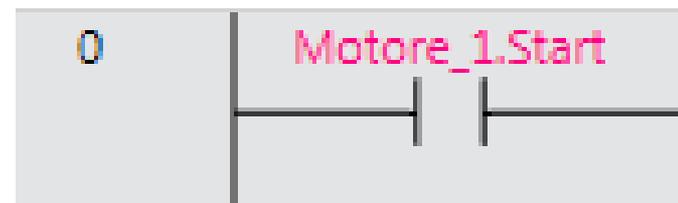
Definire tipi di dati derivati: Strutture

1. Per creare una variabile (globale o locale) di tipo struttura occorre indicare il tipo struttura precedentemente dichiarato



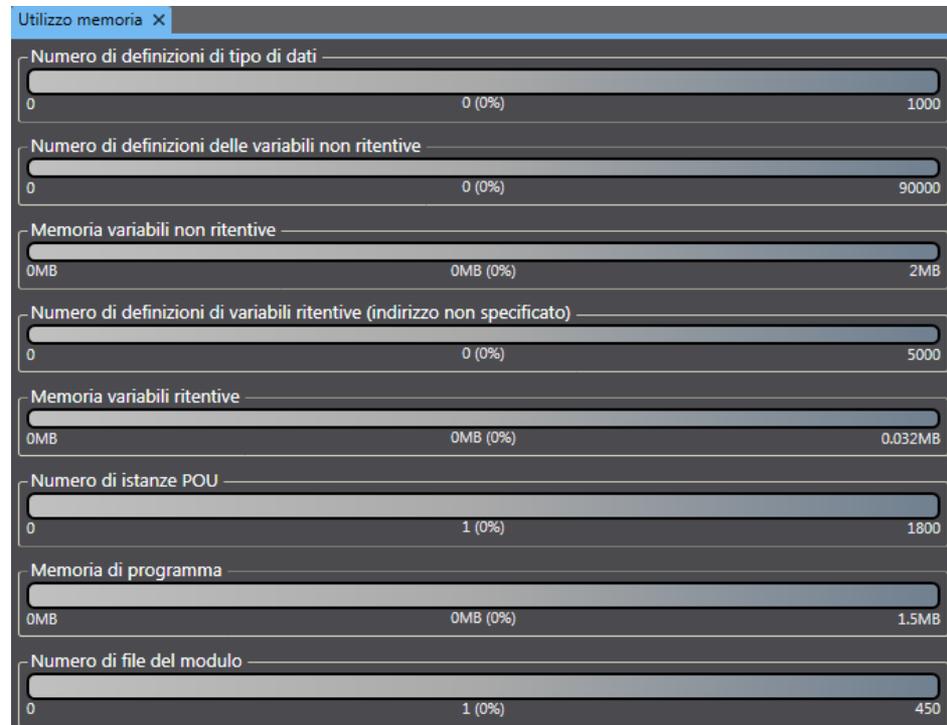
1. Per richiamare uno dei membri della struttura nel programma ladder / ST usare la sintassi:

[nome variabile].[nome membro]



Utilizzo Memoria

- Sysmac Studio permette di monitorare in qualsiasi momento la Memoria disponibile:





NX1P2: Configurazione e Impostazioni

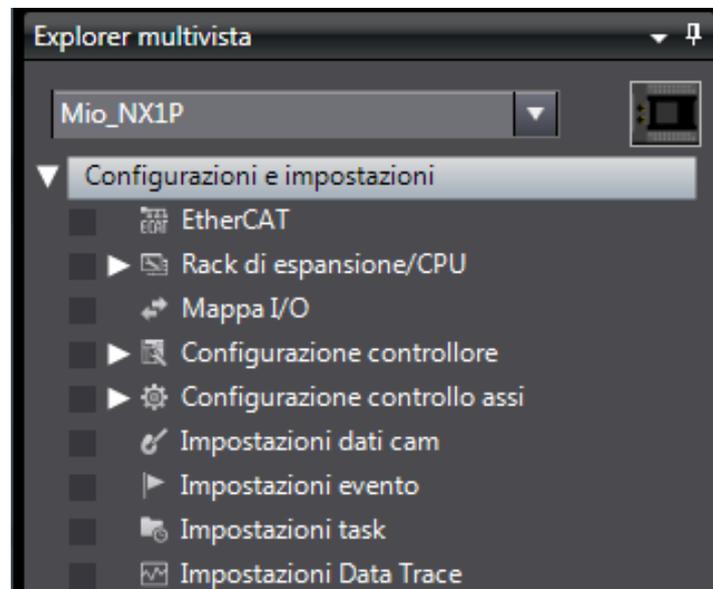
Configurare con Sysmac Studio

OMRON

OEE-I

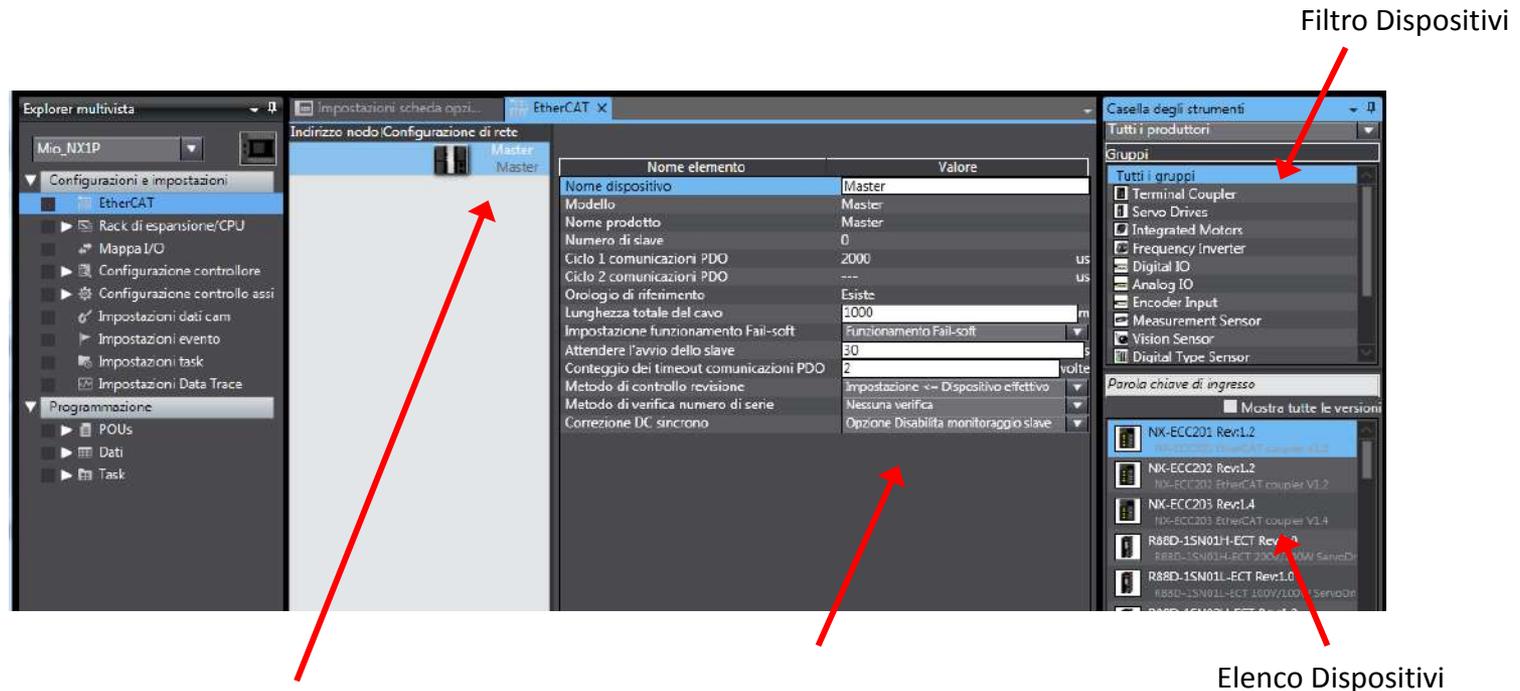
Configurazione e Impostazioni

- Come già anticipato, Sysmac Studio include una Sezione interamente dedicata alla configurazione del controllore e delle sue impostazioni:



Configurazione EtherCAT (1/2)

- Con un doppio click sulla voce EtherCAT è possibile accedere alla schermata di inserimento dei dispositivi EtherCAT che saranno collegati alla porta EtherCAT integrata nella CPU:



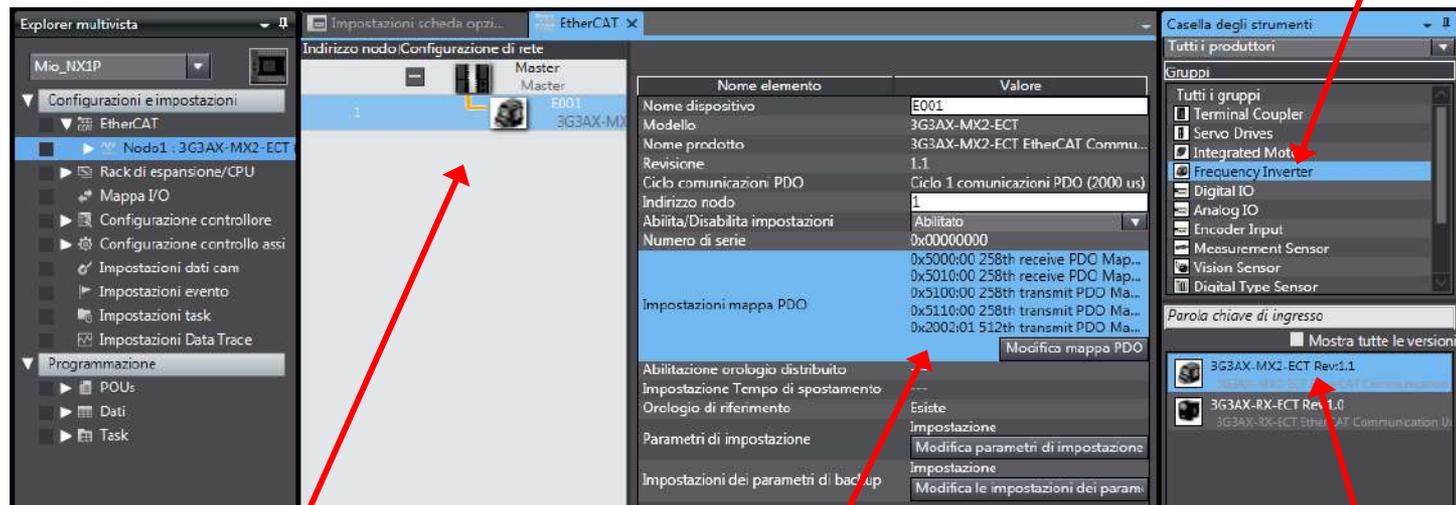
Rappresentazione grafica e selezione dei dispositivi inseriti

Proprietà dispositivo Selezionato

Elenco Dispositivi

Configurazione EtherCAT (2/2)

- Per aggiungere un dispositivo EtherCAT è sufficiente selezionarlo dall'elenco dispositivi di destra e trascinarlo nel Controllore (Master della rete) a sinistra
- Esempio di inserimento Inverter MX2



Rappresentazione grafica e selezione dei dispositivi inseriti

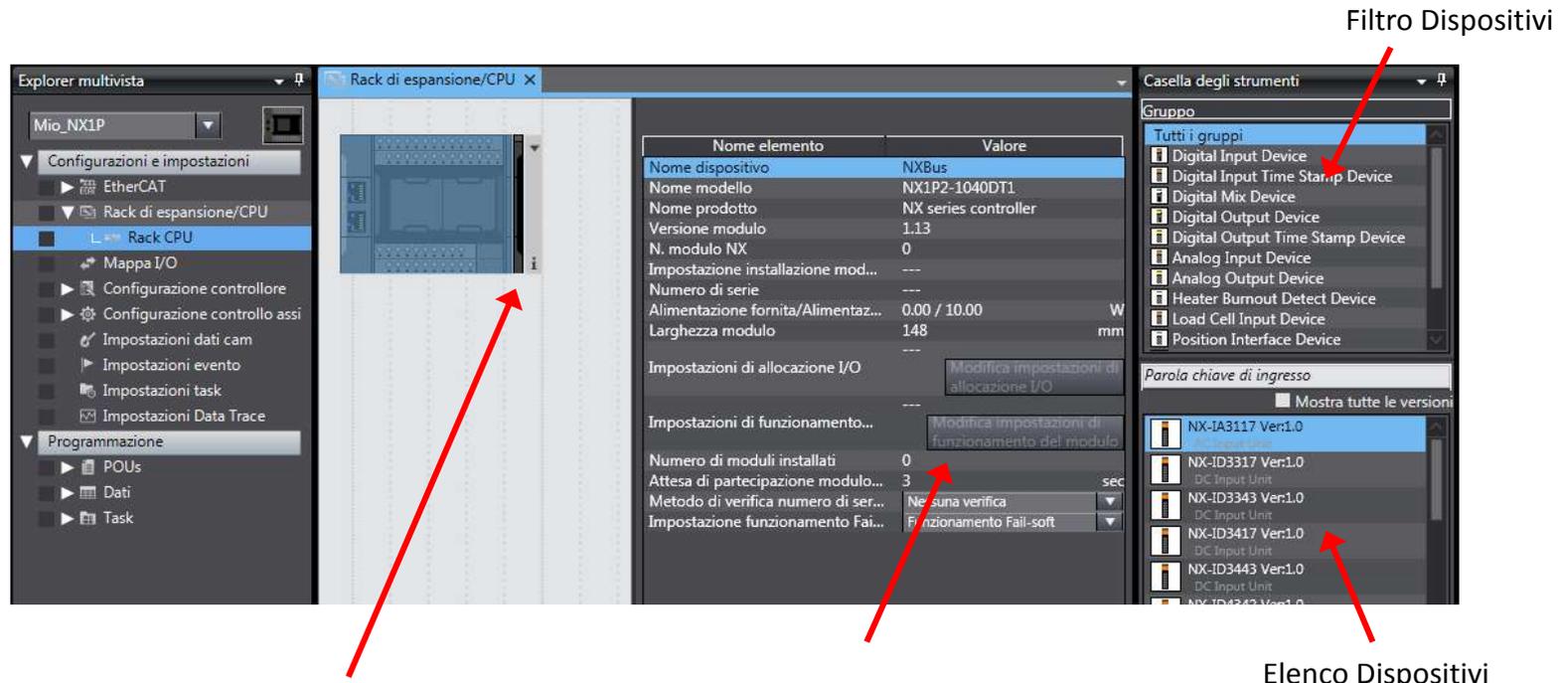
Proprietà disponibili per il dispositivo selezionato (ad esempio Modifica mappa PDO)

Filtro Dispositivi applicato

Dispositivo trascinato sul controllore

Configurazione Rack CPU (1/3)

- Con un doppio click sulla voce 'Rack CPU' è possibile accedere alla schermata di inserimento dei moduli NX I/O che saranno collegati a lato della CPU:



Rappresentazione grafica e selezione dei dispositivi inseriti

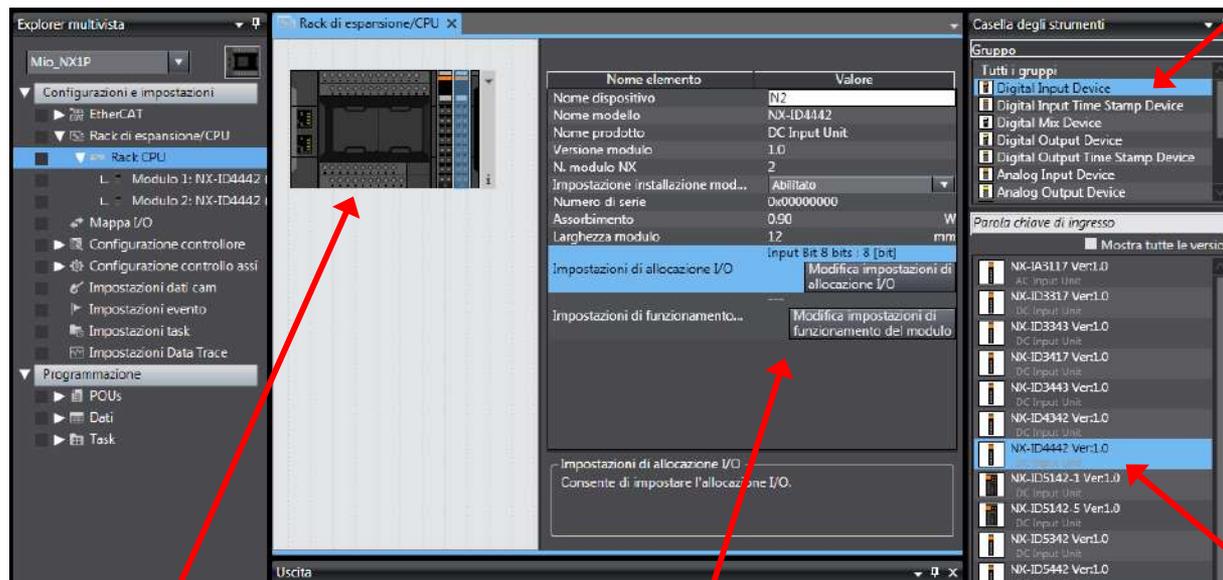
Proprietà dispositivo Selezionato

Elenco Dispositivi

Configurazione Rack CPU (2/3)

- Con lo stesso principio visto per la EtherCAT, per aggiungere un dispositivo NX I/O è sufficiente selezionarlo dall'elenco dispositivi di destra e trascinarlo a lato del Controllore
- Esempio di inserimento di due moduli NX I/O:

Filtro Dispositivi applicato



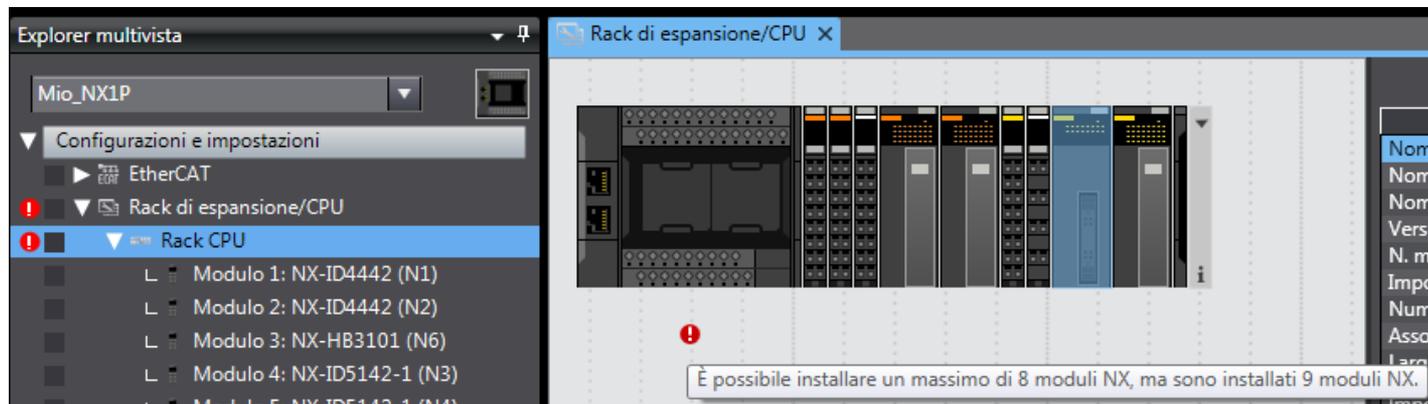
Rappresentazione grafica e selezione dei dispositivi inseriti

Proprietà disponibili per il dispositivo selezionato e indicazioni su assorbimento, ecc.

Dispositivo trascinato sul controllore

Configurazione Rack CPU (3/3)

- Il numero massimo di moduli collegabili a fianco della CPU è 8
- Nel caso in cui si inserissero più moduli il Sysmac Studio genererà un errore:



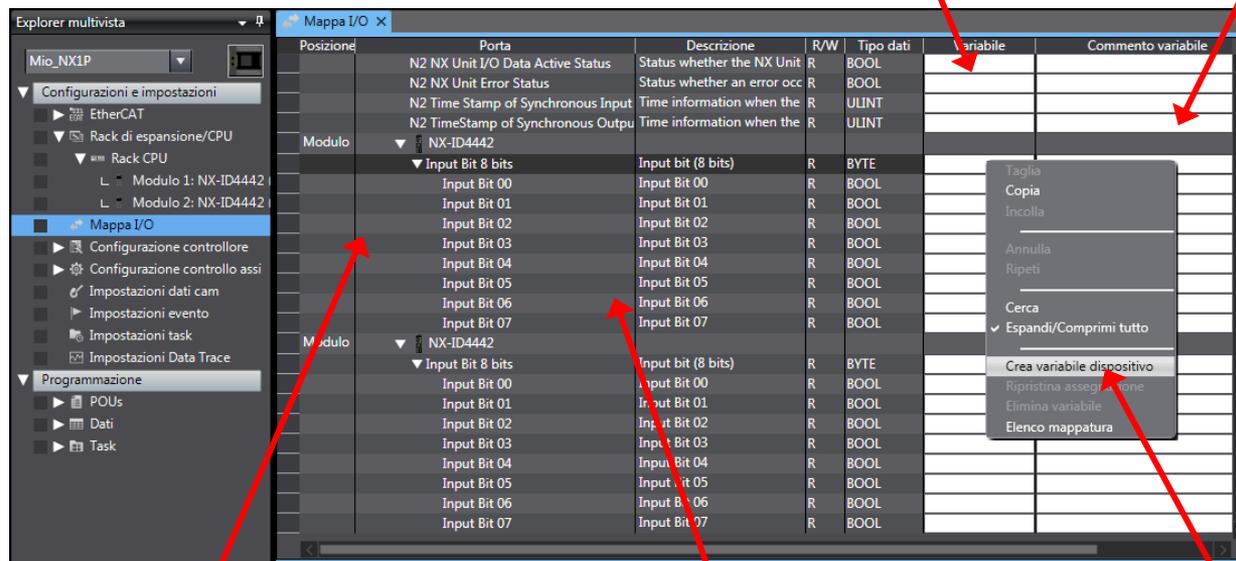
- **NOTA BENE:** Alcuni I/O necessitano di un modulo di alimentazione ausiliaria (NX-PF) sul BUS locale, che viene conteggiato come un modulo!

Configurazione Mappa I/O

- Con un doppio click sulla voce 'Mappa I/O' è possibile assegnare le Variabili Globali del controllore a ciascun I/O fisico disponibile:

Assegnazione manuale Variabili Globali agli I/O del dispositivo

Eventuale commento



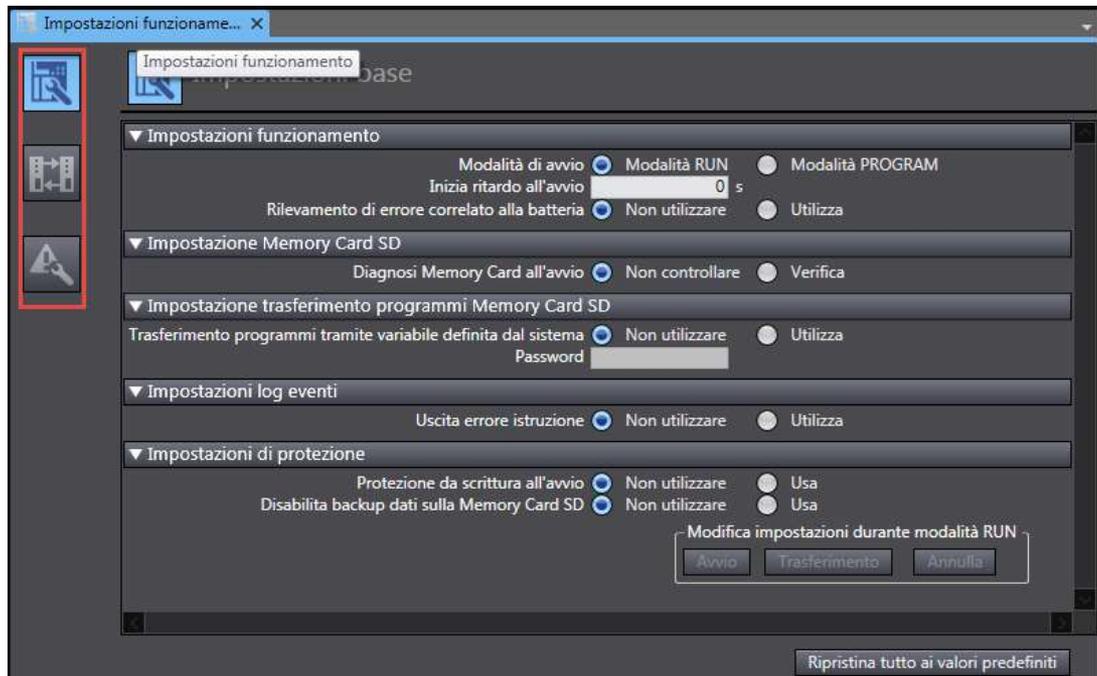
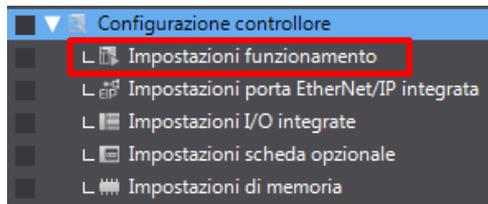
Selezione dei dispositivi locali e 'remoti' configurati

I/O Disponibili per ciascun modulo

Creazione automatica Variabili Globali associate agli I/O del dispositivo fisico

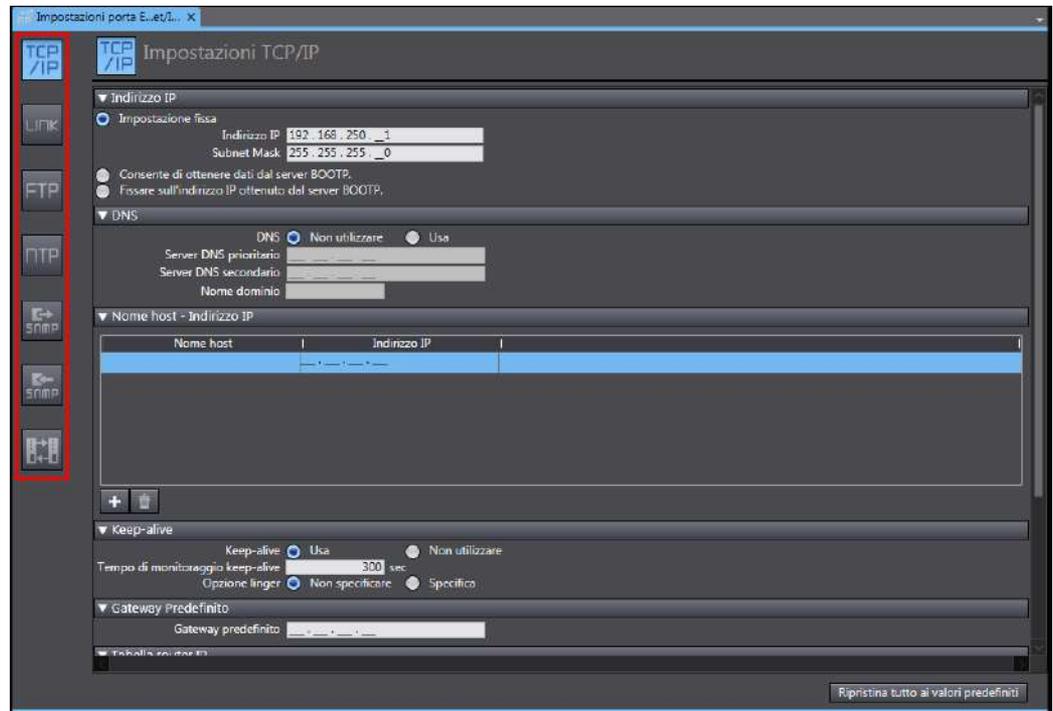
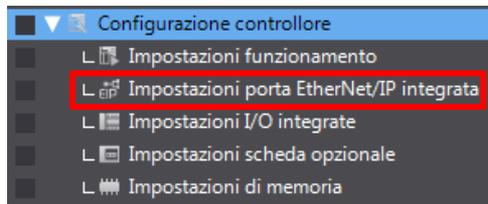
Configurazione del Controllore (1/5)

- Impostazioni Funzionamento:
 - Consentono di specificare le impostazioni di Base (ad esempio la Modalità di Avvio) e alcune impostazioni 'Avanzate' quali messaggistica FINS ed Eventi



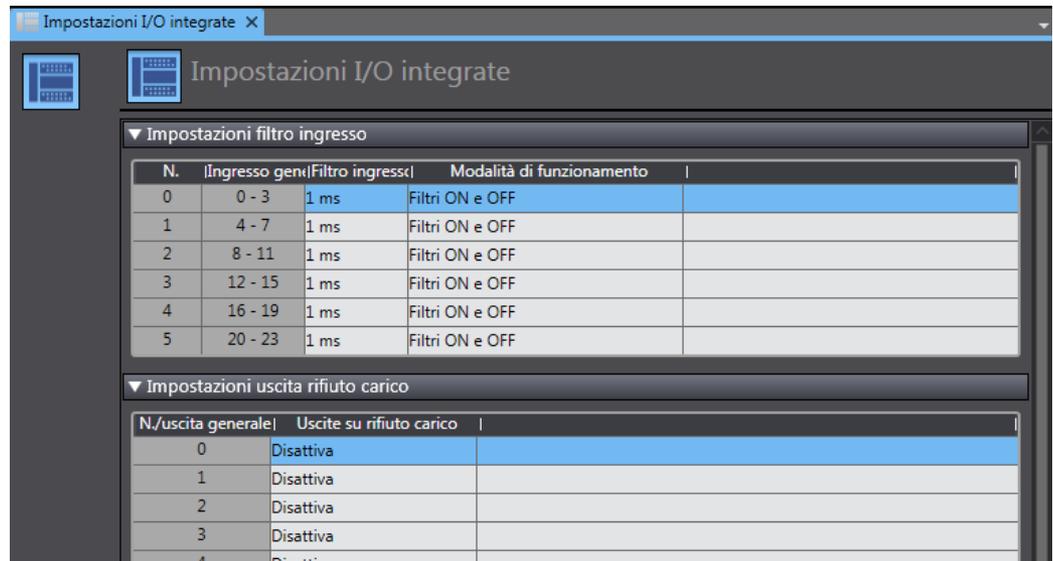
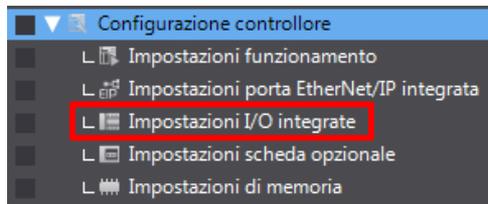
Configurazione del Controllore (2/5)

- Impostazioni Ethernet/IP Integrata:
 - Consentono di specificare le impostazioni relative alla porta Ethernet/IP integrata nella CPU, quali impostazioni TCP/IP, Data Link, FTP, FINS e altro.



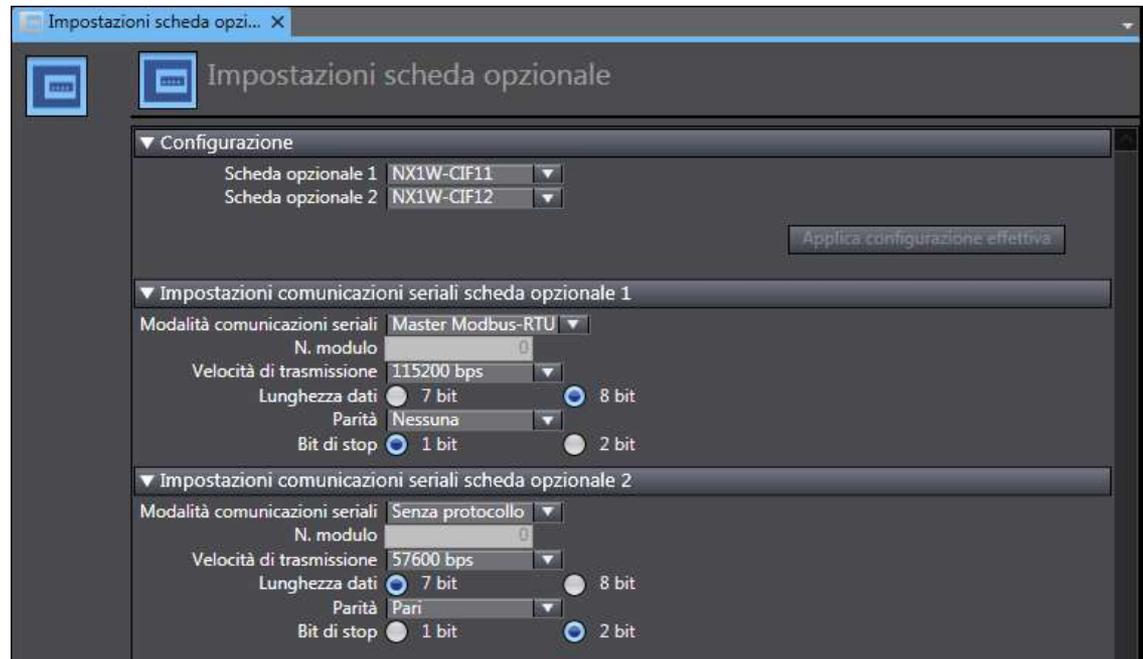
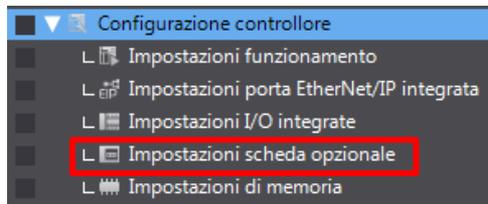
Configurazione del Controllore (3/5)

- Impostazioni I/O Integrati:
 - Consentono di impostare filtri sugli ingressi integrati dell'NX1P2 e la modalità di funzionamento delle Uscite in caso di CPU in STOP.



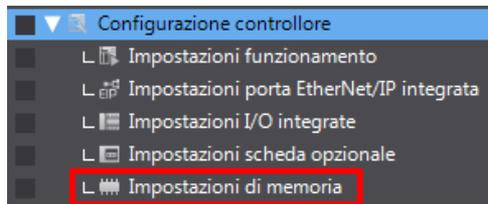
Configurazione del Controllore (4/5)

- Impostazioni Scheda Opzionale:
 - Consentono di impostare i dettagli sui protocolli utilizzati dalle schede opzionali di comunicazione.



Configurazione del Controllore (5/5)

- Impostazioni di Memoria:
 - Consentono di abilitare le aree dati (e la relativa dimensione) utilizzate per l'eventuale comunicazione di dispositivi collegati al PLC tramite protocollo FINS.



Impostazioni di memoria X

Impostazioni di memoria per i moduli serie CJ

Attiva	Tipo di area	Dimensione (numero di canali)	Ritentiva
<input type="checkbox"/>	CIO	5144	Non ritentiva
<input checked="" type="checkbox"/>	WR	512	Non ritentiva
<input type="checkbox"/>	HR	512	Mantenuta
<input checked="" type="checkbox"/>	DM	4096	Mantenuta



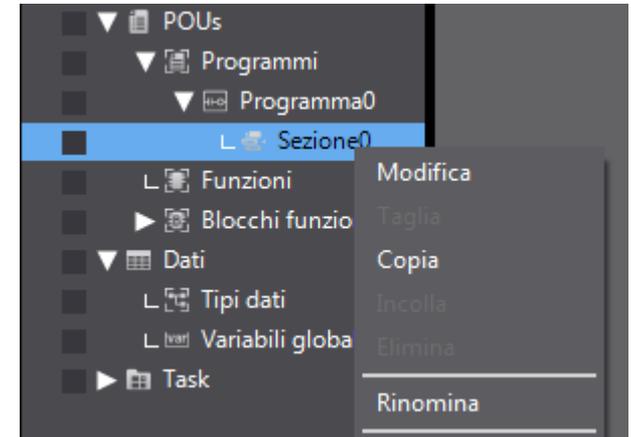
Sysmac Studio: Creazione di un programma

Programmare con Sysmac Studio

OMRON

Programma di default

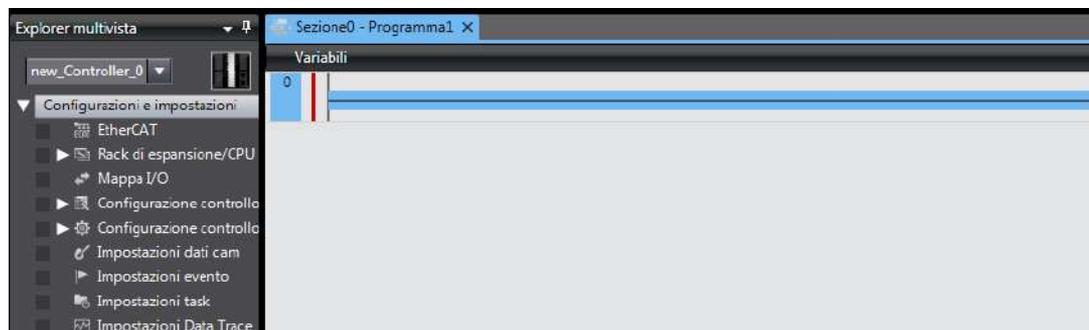
Alla creazione di un nuovo progetto viene creato in automatico un nuovo programma ladder nel percorso Programmazione -> POU's -> Programmi, chiamato **Programma0** con una sezione chiamata **Sezione0**



Per rinominare il programma cliccare col tasto destro del mouse e quindi premere su **Rinomina**;

Procedere allo stesso modo per rinominare la sezione;

Aperto la Sezione0 con un doppio clic si visualizzerà l'editor con una prima rung vuota



Verifica del programma

La verifica del programma trova gli errori di sintassi presenti in tutti i POU dichiarati

Ci sono due possibili verifiche:

- Verifica di tutti i programmi
- Verifica dei programmi selezionati

Progetto	Unità di controllo	Simulazione
Verifica tutti i programmi		F7
Verifica programmi selezionati		Shift+F7
Compila unità di controllo		F8
Ricrea unità di controllo		

- Cliccare su **Verifica tutti i programmi*** o **Verifica i programmi selezionati*** nel menu **Progetto**.



(*: Si possono anche premere F7 e la combinazione F7+Shift)

Compilazione del programma

La **compilazione** è il processo che produce il programma scaricabile nell'NJ.

La compilazione non viene completata se vengono trovati errori.

Eventuali errori trovati verranno mostrati nel **Pagina tab di compilazione** nel **Link Pane:**

Dopo la prima compilazione vengono compilate solo le **modifiche**.

Per effettuare di nuovo una compilazione completa occorre cliccare su «**Ricrea unità di controllo**»

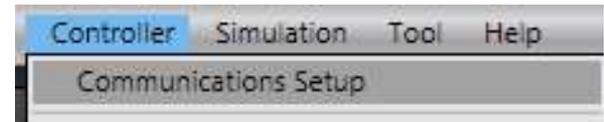
Per compilare / ricompilare il programma cliccare su **Compila unità di controllo** o su **Ricrea unità di controllo**, sotto il menu **Progetto**

(*: Si può anche usare il tasto F8)

Progetto	Unità di controllo	Simulazione
	Verifica tutti i programmi	F7
	Verifica programmi selezionati	Shift+F7
	Compila unità di controllo	F8
	Ricrea unità di controllo	

Tipo di connessione

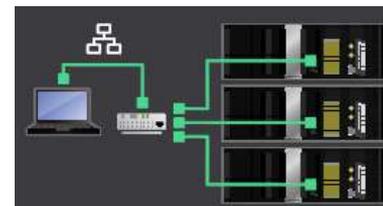
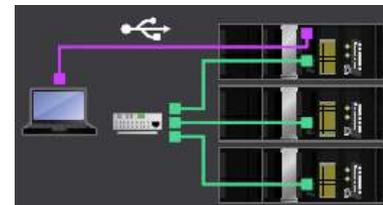
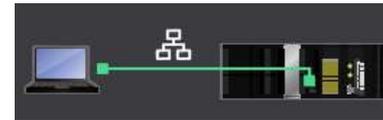
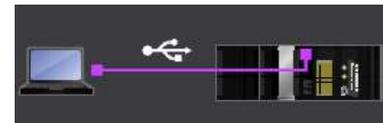
Per aprire la finestra di collegamento cliccare su **Impostazioni di comunicazione** sotto il menu **Unità di controllo**



- **Menu tipo di connessione**

Ci sono quattro possibili modalità di collegamento selezionabili:

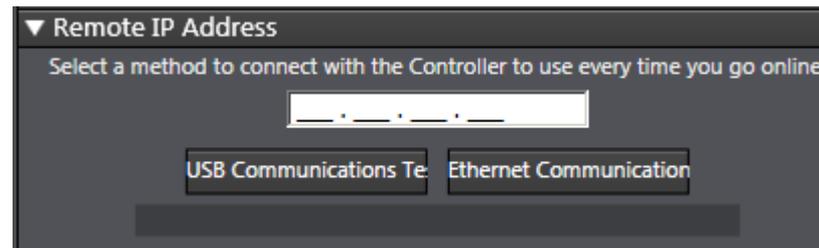
- **Connessione diretta via USB:** La porta USB del computer è collegata direttamente alla porta USB dell'NJ.
- **Connessione diretta via Ethernet:** La porta Ethernet del computer è collegata direttamente alla porta Ethernet/IP dell'NJ.
- **Connessione remota via USB:** La porta USB del computer è collegata direttamente ad un NJ, il quale è collegato tramite uno switch ethernet all'NJ a cui si vuole accedere.
- **Connessione Ethernet via hub:** La porta Ethernet del computer è collegata ad un hub a cui è collegato l'NJ a cui si vuole accedere.



Indirizzo IP e Test

- **Indirizzo IP remoto**

Imposta l'indirizzo IP dell'NJ a cui ci si vuole connettere



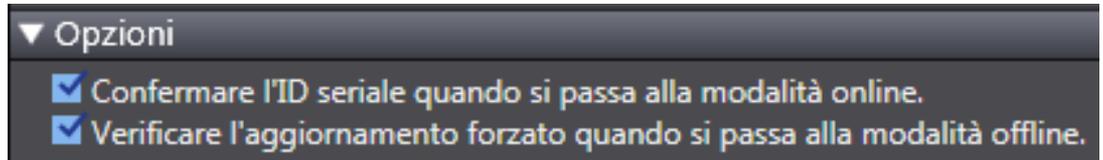
- **Test comunicazioni USB and Test comunicazioni Ethernet:**

Esegue un test di comunicazione tra il Sysmac Studio e l'NJ collegato tramite USB oppure via Ethernet all'indirizzo IP indicato. Il risultato sarà mostrato nel campo sottostante:

- se va a buon fine: *“Test OK”*
- In caso di errore: *“Si è verificato un errore di comunicazione”*.

Opzioni del collegamento

- **Opzioni**

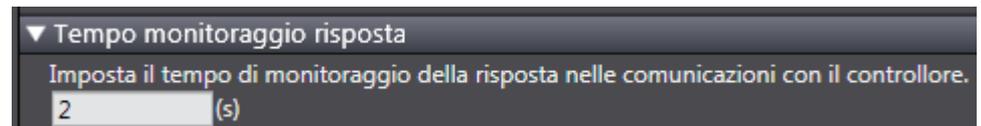


Confermare l'ID seriale quando si passa alla modalità in linea: Se questa opzione è selezionata, ogni qual volta si esegue il collegamento vengono verificati il nome del progetto e il serial ID dell'NJ

Verificare l'aggiornamento forzato quando si passa alla modalità non in linea: Se questa opzione è selezionata, quando si torna offline viene verificata la presenza di eventuali forzature di variabili nel programma.

- **Tempo monitoraggio risposta**

Impostare il timeout di comunicazione

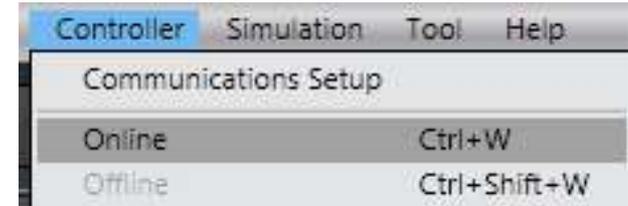


Se non si riceve risposta entro il tempo impostato, verrà mostrato un errore

Online

Due possibilità:

1. Selezionare **In linea** nel menu **Unità di controllo**
2. Cliccare l'icona **In linea** nella toolbar.



Una volta in linea, l'editor del Sysmac Studio presenterà una barra orizzontale gialla

Disconnessione (offline)

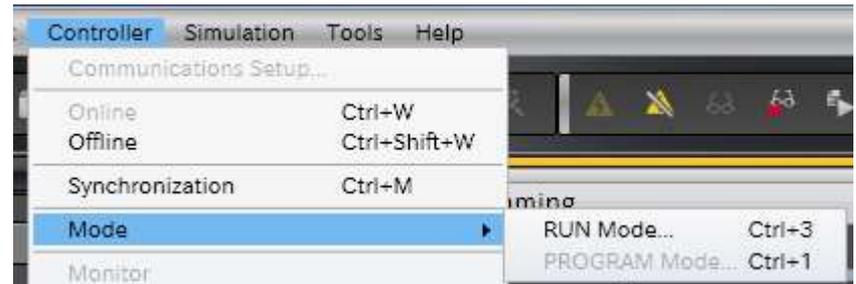
Due possibilità:

1. Selezionare **Non in linea** dal menu **Unità di controllo**
2. Cliccare l'icona **Non in linea** nella toolbar.

Modalità operativa

Per cambiare la modalità operativa dell'NJ:

1. Cliccare **Modalità** dal menu **Unità di controllo**
2. Selezionare **RUN** o **PROGRAM**

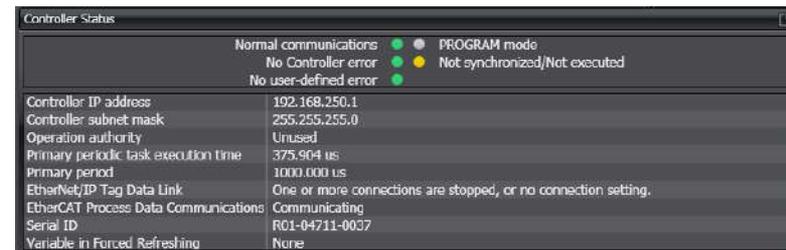


Oppure:

Cliccare l'icona **RUN** o **PROGRAM** sulla toolbar

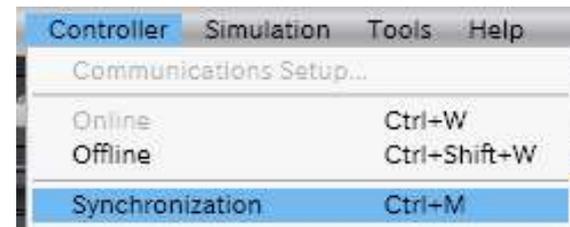


NOTA: La modalità operativa corrente dell'NJ è mostrata sul **Controller Status Pane**, nell'angolo in basso a destra.



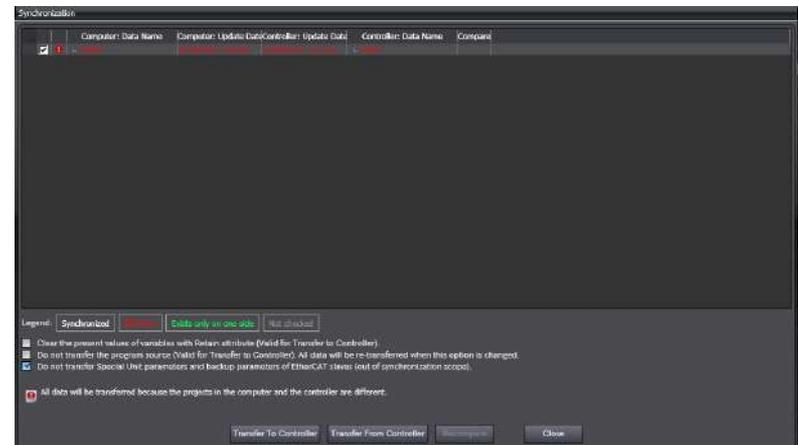
Sincronizzazione

La sincronizzazione consente il download/upload del programma utente sull'NJ, per garantire che i dati sul Sysmac Studio e sull'NJ corrispondano.



Per eseguire la sincronizzazione:

1. Seleziona **Sincronizzazione** dal menu **Unità di controllo** oppure cliccare l'icona **Sincronizzazione** sulla toolbar;
2. Cliccare sul pulsante **Trasferisci all'unità di controllo** o **Trasferisci dall'unità di controllo**;



Sincronizzazione: legenda colori

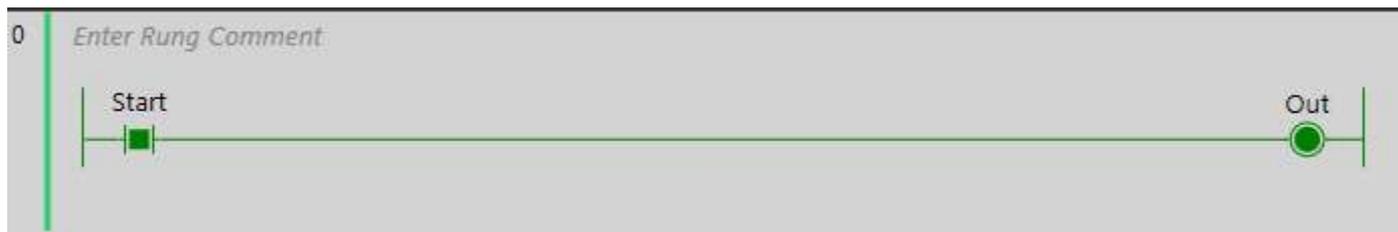
Colore del testo	Stato	Descrizione
Bianco	Sincronizzato	I dati su Sysmac Studio e NJ corrispondono.
Rosso	Differente	<p>I dati su Sysmac Studio e NJ non sono gli stessi. I seguenti casi sono trattati come "diversi".</p> <ul style="list-style-type: none"> -Se uno o più file designati come dati sincronizzati non sono uguali -Se i nomi dei dati sincronizzati non sono uguali -Se l'ordine di visualizzazione dei dati sincronizzati non è lo stesso (tuttavia, se l'ordine di visualizzazione non è lo stesso, i dati vengono visualizzati in ordine Sysmac Studio). -Se gli elementi di dati sincronizzati non sono uguali.
Verde	Presente solo su un lato	I dati esistono solo sul Sysmac Studio.
Blu	Presente solo su un lato	I dati esistono solo sull'NJ.
Grigio	Non verificato	<p>Il programma non è compilato o si è verificato un errore.</p> <p>Il parametro è relativo ad un modulo speciale CJ o ad uno slave EtherCAT.</p>

Monitoraggio

Il monitoraggio può essere effettuato dall'editor Ladder o ST, dalla Scheda monitoraggio o dalla Mappa I/O.

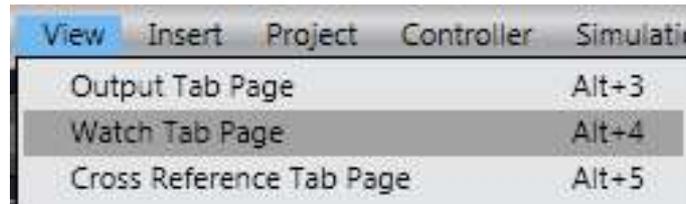
Per monitorare tramite l'editor:

Cliccare su Monitoraggio dal menu Controllore (attivo di default).



Monitoraggio: Finestra di Monitoraggio

Selezionare su *Scheda monitoraggio* dal menu *Visualizza*.

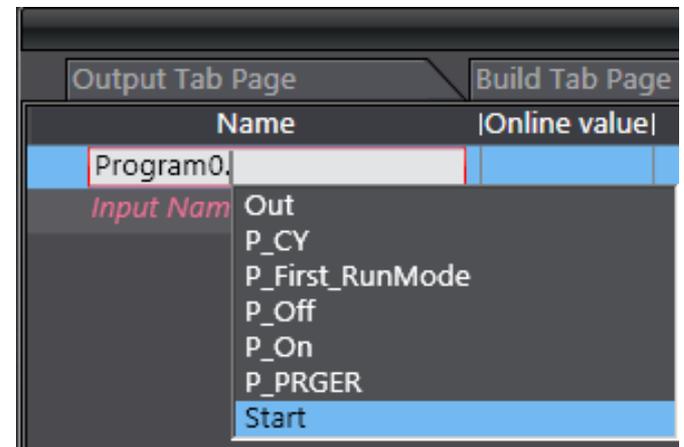


La scheda di monitoraggio verrà mostrata nel **Link Pane**:



Per aggiungere una variabile da monitorare:

- Cliccare su “*Nome ingresso...*” nel campo **Nome**;
- La sintassi è:
[Nome programma].[Nome variabile]



Monitoraggio: Finestra di Monitoraggio

Name	Online value	Modify	Data type	Address	Data format
Program0.Start <i>Input Name:</i>	False	TRUE FALSE	BOOL		Boolean ▾

Nome della
variabile

Valore
attuale

Modifica
valore
attuale

Tipo dati

Percorso
fisico

Rappresentazione

Monitoraggio: Mappa I/O

Si possono monitorare i valori attuali delle variabili dispositivo tramite il campo **Valore**, mentre si è in linea con l'NJ.

- Selezionare **Mappa I/O** nella vista **Configurazione**.
- Monitorare i valori attuali delle variabili tramite il campo «Valore»

Position	Port	Description	R/W	Data Type	Value	Variable
CPU/Expansion Racks						
CPU Rack 0						
[00]						
CJ1W-SCU32 (Serial Communic						
Com_UnitSta						
	Com_UnitLogMemErr	Serial Communication Unit St	R	WORD	16#0	J01_Com_UnitSta
	Com_UnitPmrDatErr	Error Log EEPROM Error	R	BOOL	FALSE	J01_Com_UnitLogMemErr
		Protocol Data Error	R	BOOL	FALSE	J01_Com_UnitPmrDatErr
	P1_PortCfg	Port1: Port Settings	RW	WORD	16#0	J01_P1_PortCfg
	P1_SerSetCfg	Port1: User-specified Setting I	RW	BOOL	FALSE	J01_P1_SerSetCfg
	P1_StartBitCfg	Port1: Start Bits	RW	BOOL	FALSE	J01_P1_StartBitCfg
	P1_DatBitCfg	Port1: Data Length	RW	BOOL	FALSE	J01_P1_DatBitCfg
	P1_StopBitCfg	Port1: Stop Bits	RW	BOOL	FALSE	J01_P1_StopBitCfg
	P1_ParityYNCfg	Port1: Parity	RW	BOOL	FALSE	J01_P1_ParityYNCfg
	P1_ParityBitCfg	Port1: Parity Even/Odd	RW	BOOL	FALSE	J01_P1_ParityBitCfg
	P1_BaudrateCfg	Port1: Baud Rate	RW	USINT	0	J01_P1_BaudrateCfg
	P1_SendDelayCfg	Port1: Send Delay Settings	RW	WORD	16#0	J01_P1_SendDelayCfg
	P1_SendDelaySetCfg	Port1: Send Delay Time	RW	BOOL	FALSE	J01_P1_SendDelaySetCfg
	P1_HlkCfg	Port1: Host-Link Protocol Set	RW	WORD	16#0	J01_P1_HlkCfg

Monitor type: Data type Binary Hex Signed decimal Unsigned decimal

Bit order: MSB-LSB LSB-MSB

Modifica online

Per iniziare la modifica in linea:

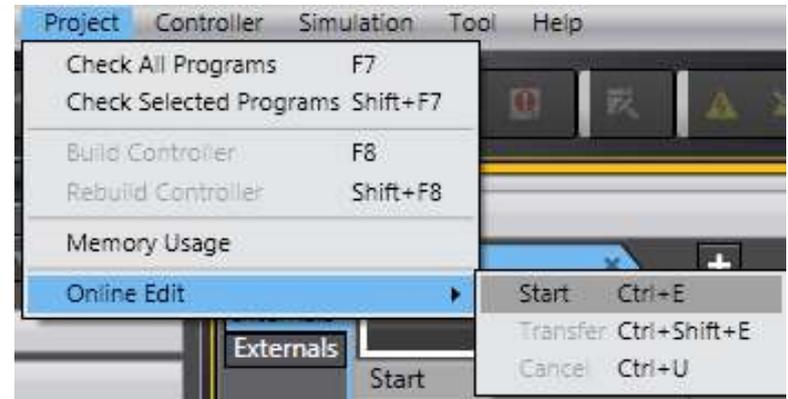
1. Selezionare **Modifica in linea** dal menu **Progetto**;
2. Cliccare **Avvia**.

Oppure:

Premere la combinazione CTRL + E.

Terminate le operazioni di modifica:

1. Selezionare **Modifica in linea** dal menu **Progetto**;
 2. Cliccare **Trasferisci**.
- Oppure:
Premere la combinazione CTRL + SHIFT + E.



Mappatura tastiera

Sono fornite le seguenti scorciatoie da tastiera:

Operazione	Scorciatoia da tastiera
Inserimento di contatto	C
Inserimento di contatto negato	/ (Shift + `)
Inserimento di un contatto in parallelo	W
Inserimento di un contatto negato in parallelo	X
Inserimento di una bobina	O
Inserimento di una bobina negata	Q
Inserimento di un blocco funzione	F
Inserimento di una funzione	I
Verifica di tutti i programmi	F7
Verifica dei programmi selezionati	Shift + F7
Compilazione del progetto	F8

Mappatura tastiera

Sono fornite le seguenti scorciatoie da tastiera:

Operazione	Scorciatoia da tastiera
Collegamento in linea	Ctrl + W
Disconnessione (non in linea)	Ctrl + Shift + W
Commuta in modalità PROGRAM	Ctrl + 1
Commuta in modalità RUN	Ctrl + 3
Sincronizza	Ctrl + M
Forza a TRUE una variabile booleana	Ctrl + J
Forza a FALSE una variabile booleana	Ctrl + K
Rimuovi la forzatura da una variabile booleana	Ctrl + L
Inizia la sessione di modifica online	Ctrl + E
Trasferisci le modifiche effettuate online	Ctrl + Shift + E
Non trasferire le modifiche effettuate	Ctrl + U



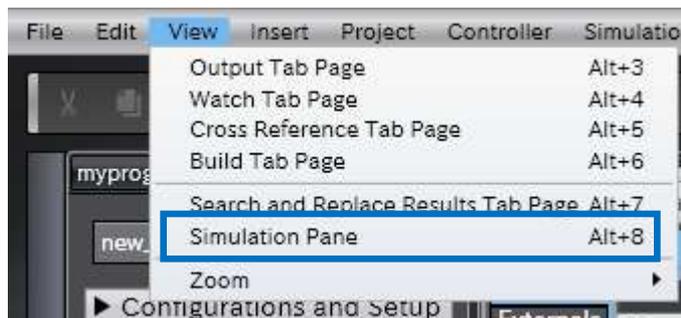
Sysmac Studio: Simulatore

Uso del simulatore

Due possibilità:

Simulazione		Strumenti	Guida
Esegui			F5
Pausa			Ctrl+Alt+Break
Arresta			Shift+F5
Esecuzione step			F10

1. Dal menu **Simulazione**, selezionare **Esegui**
2. Dal menu **Visualizzazione** selezionare **Pannello Simulatore** (oppure premere Alt+F8), quindi premere il pulsante **RUN** nella pannello simulatore che apparirà in basso a destra



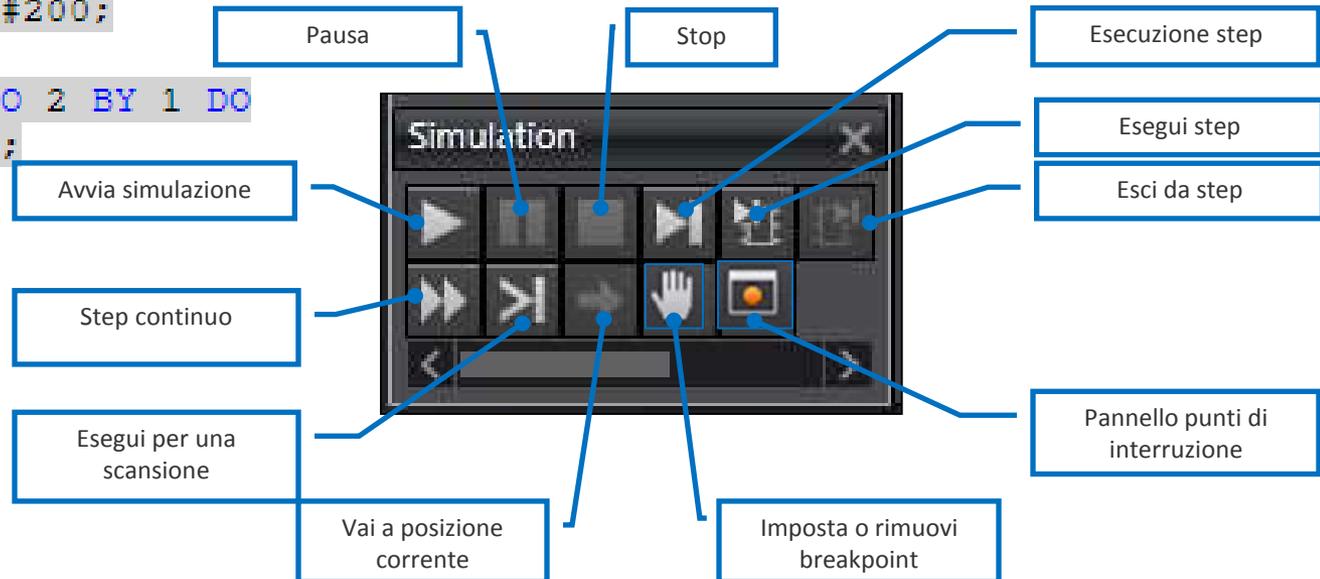
Impostare punti di interruzione e eseguire il programma per passi

Per semplificare il debug del programma è possibile eseguire il programma per passi o per punti di interruzione.

```

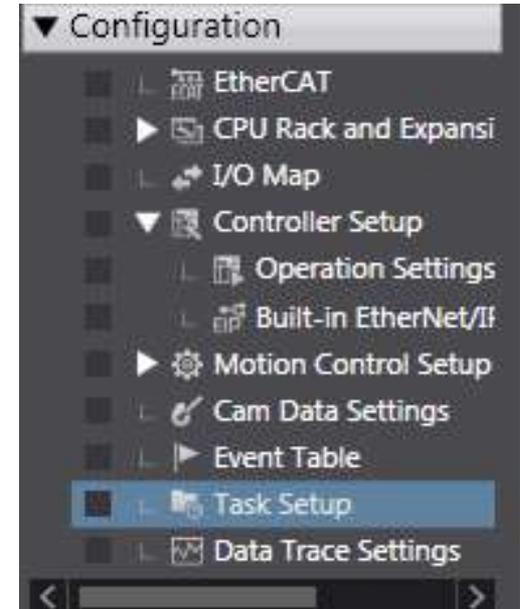
1  varA:=13;
2  varB:=49;
3  IF varA < DINT#1 THEN
4      varA:=DINT#100;
5  END_IF;
6  IF varA > DINT#12 THEN
7      varA:=DINT#200;
8  END_IF;
9  FOR varB:= 0 TO 2 BY 1 DO
10     varA:=varB;
11 END_FOR;
12

```



Simulazione del tempo di esecuzione

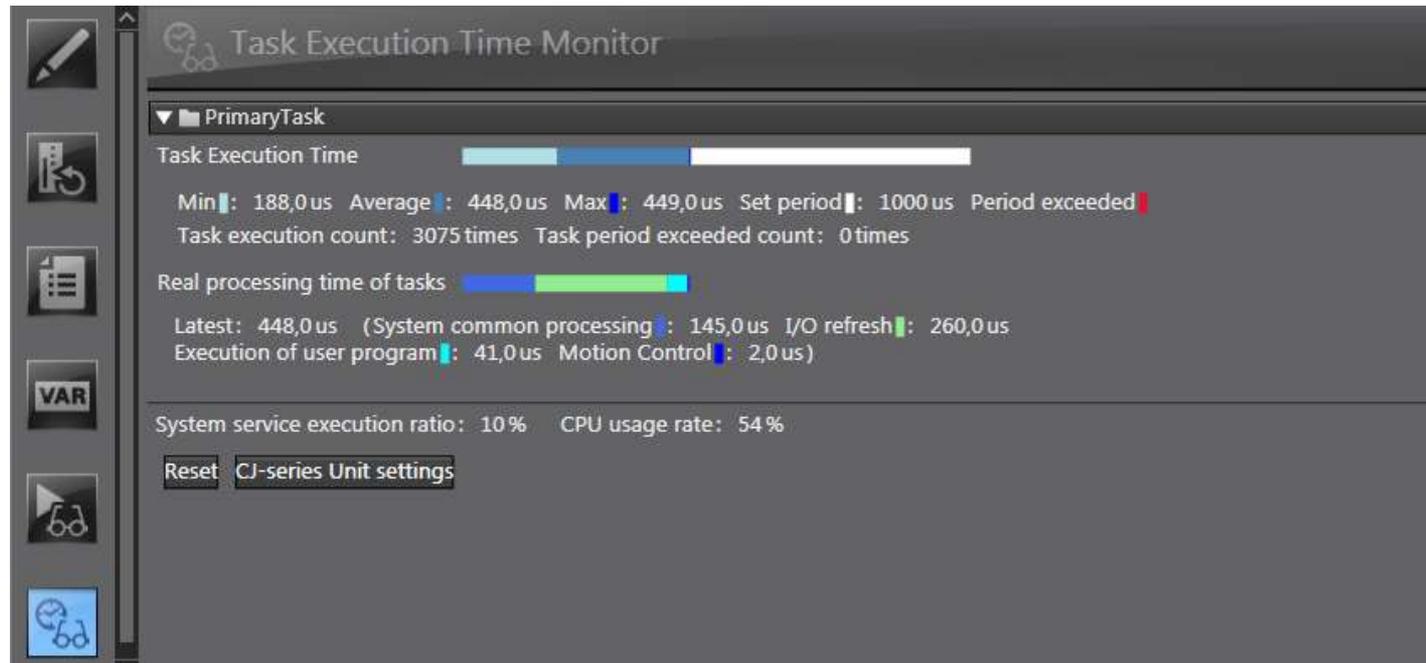
- Il simulatore è capace di calcolare con accuratezza quale sarà il tempo di esecuzione del programma utente sull'NJ reale
- Dopo aver effettuato una calibrazione del computer, il simulatore è capace di calcolare i tempi minimo, massimo e medio di esecuzione del programma.



1. Nell'*Explorer multivista*, doppio click su **Impostazioni Task** nella vista **Configurazione**
2. Quindi cliccare sull'icona **Monitoraggio tempo di esecuzione task**



Simulazione del tempo di esecuzione



- Il tempo di esecuzione simulato tiene anche conto delle interruzioni dei task a priorità più alta e dei servizi di sistema.
- Possono anche essere monitorati I cambiamenti nei tempi di esecuzione provocati dalla modifica delle condizioni del programma utente
- Vengono calcolati I tempi minimo, massimo e medio per evidenziare gli eventuali task che sfiorano I tempi di esecuzione impostati.



DOMANDE?

Sysmac Studio

Automation Software

Version 1.17

© Copyright OMRON Corporation 2011-
All Rights Reserved.
This Program is protected by U.S.
and international copyright laws.

Ladder Diagram

Corso Base Machine Controller

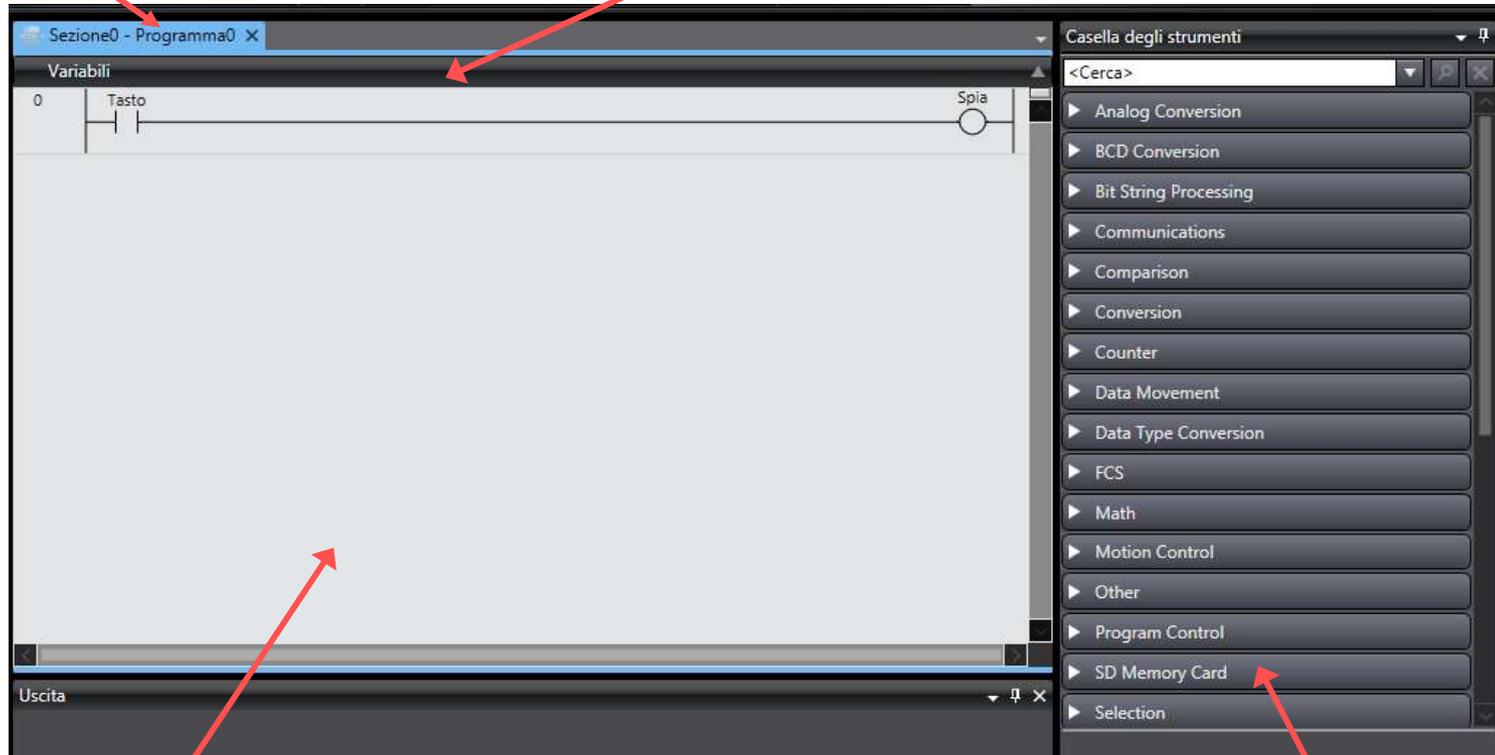
OMRON

OEE-I

Editor

Nome Sezione e Programma

Variabili e Namespace

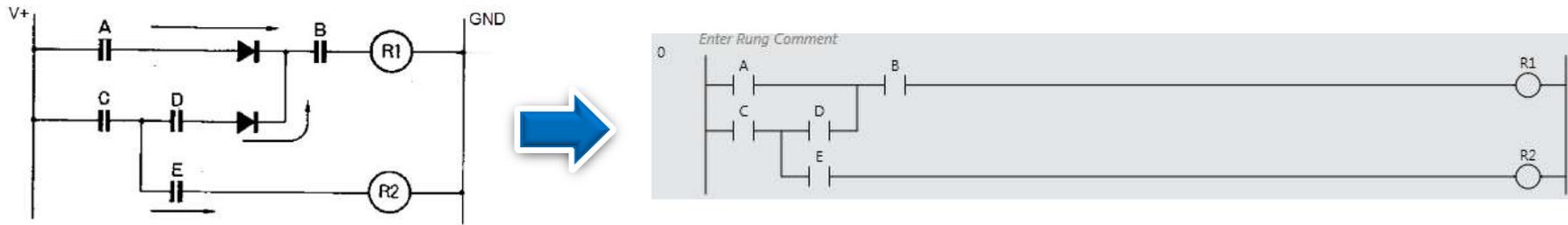


Area di lavoro

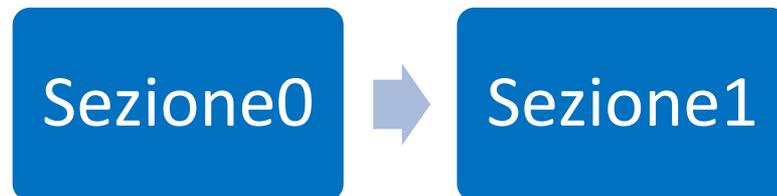
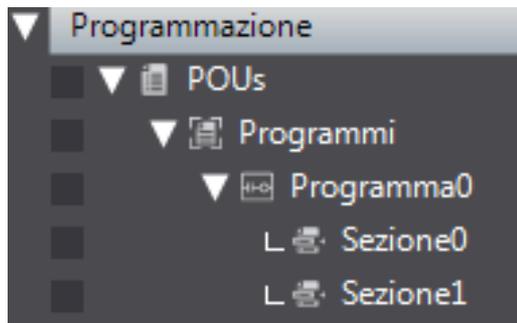
Funzioni e Blocchi
Funzione

Le Rung o 'Network'

Il linguaggio ladder proviene dai diagrammi a relè della logica cablata.

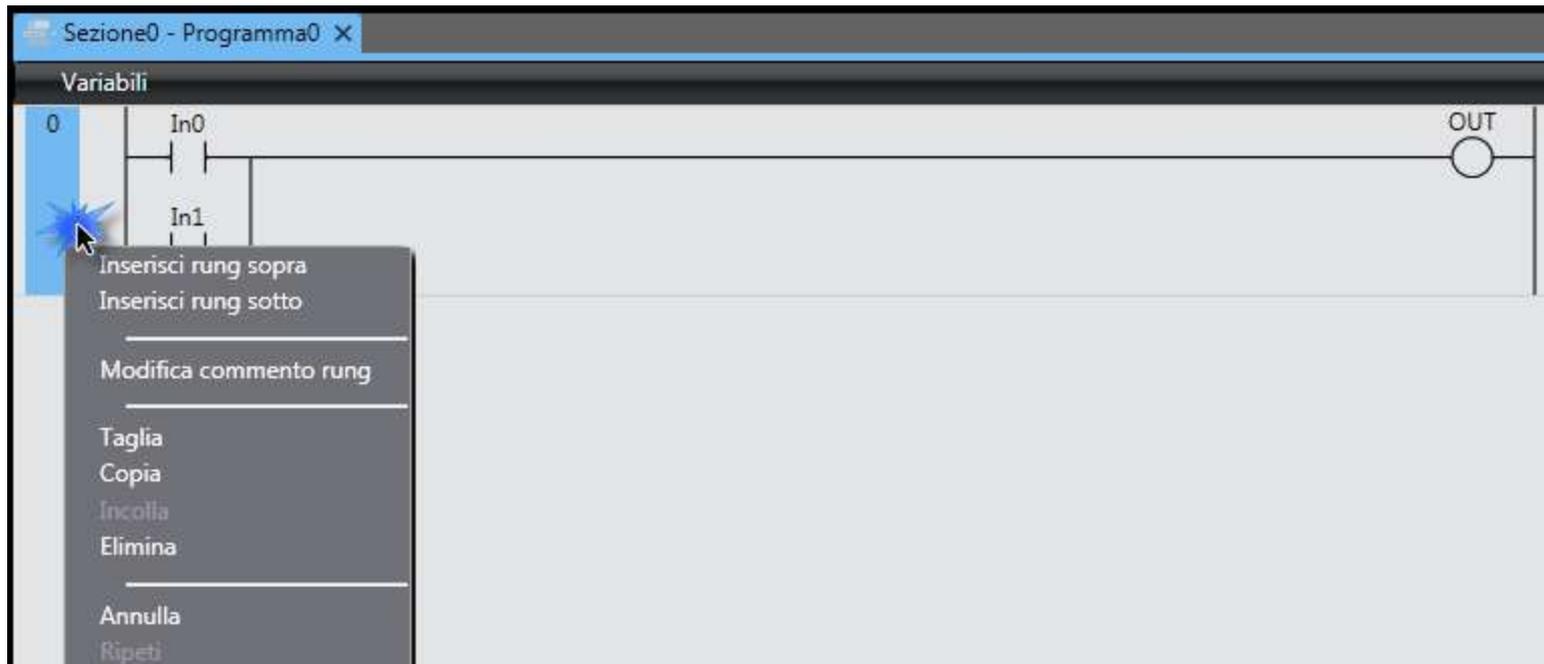


Il Programma può contenere una o più Sezioni, che sono eseguite in maniera sequenziale



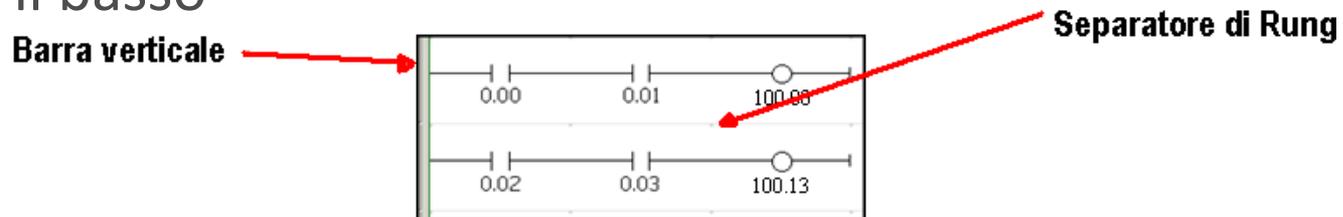
Le Rung o 'Network'

Per inserire una nuova è sufficiente selezionare con il tasto destro del mouse il bordo sinistro del Ladder e scegliere se inserire la nuova Rung al di sotto o al di sopra di quella correntemente selezionata



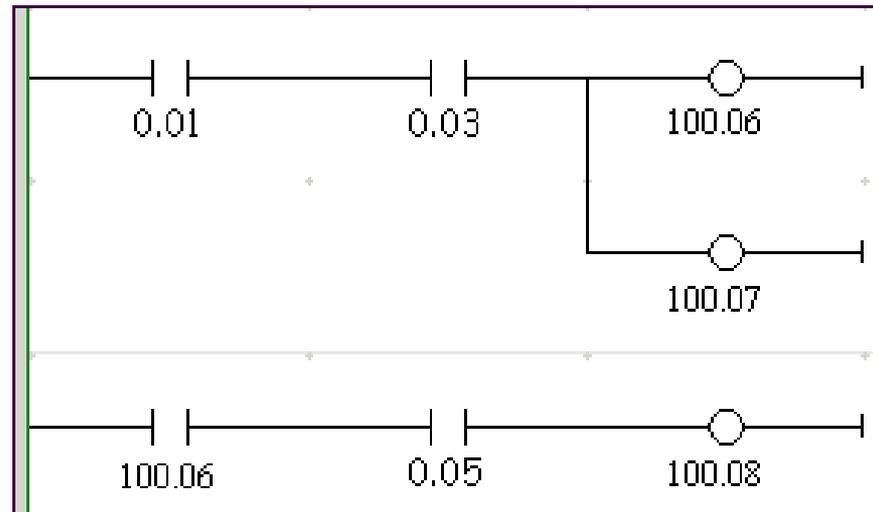
Concetti base

- Un programma in diagramma a relè (Ladder Diagram), è formato da una serie di righe circuitali
- Una riga circuitale (Network o Rung) è composta da:
 - una serie di contatti collegati tra loro in serie e/o parallelo
 - eventuali diramazioni
 - bobine o funzioni
- Ogni Rung (o Network) ha origine da una barra verticale posta a sinistra del diagramma
- Il flusso dei contatti viene letto da sinistra a destra e dall'alto verso il basso



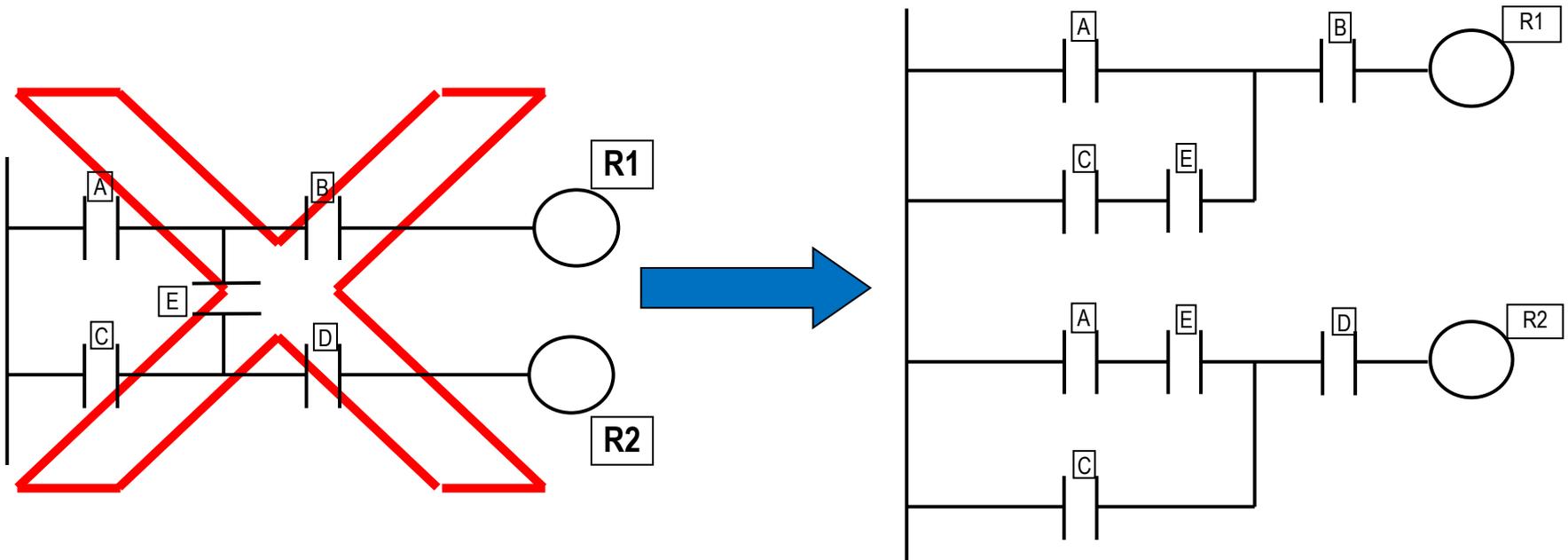
Concetti base

- E' consigliabile non programmare un contatto di uscita come bobina più di una volta
- E' possibile utilizzare come input ausiliario un contatto associato allo stato dell'uscita
- E' possibile collegare in parallelo e in serie due o più bobine



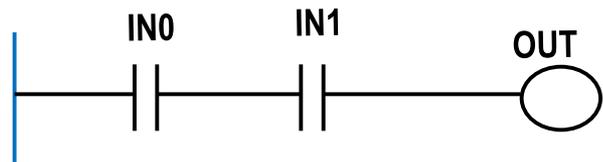
Concetti base

- Non è possibile programmare dei contatti sui rami di collegamento verticale

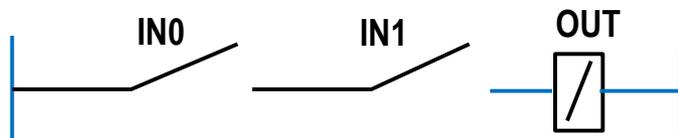


L'istruzione AND

DIAGRAMMA A RELE'

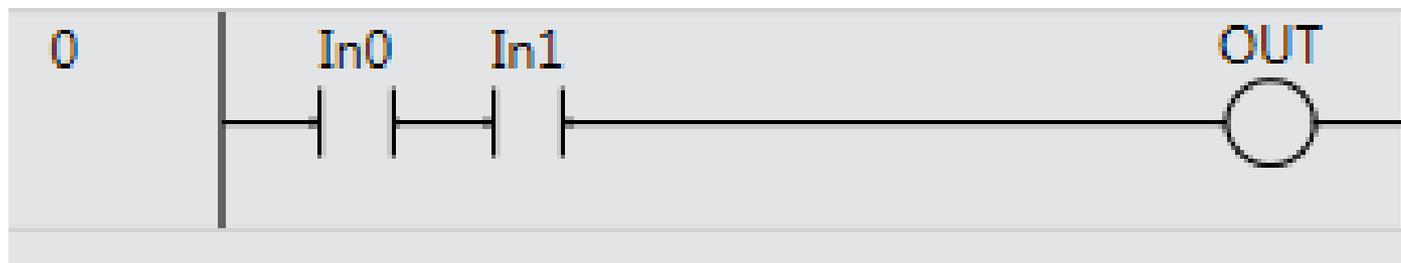


SCHEMA ELETTRICO



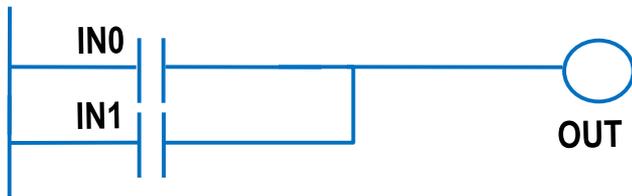
IN0	IN1	OUT
0	0	0
0	1	0
1	0	0
1	1	1

- Esempio



L'istruzione OR

DIAGRAMMA A RELE'

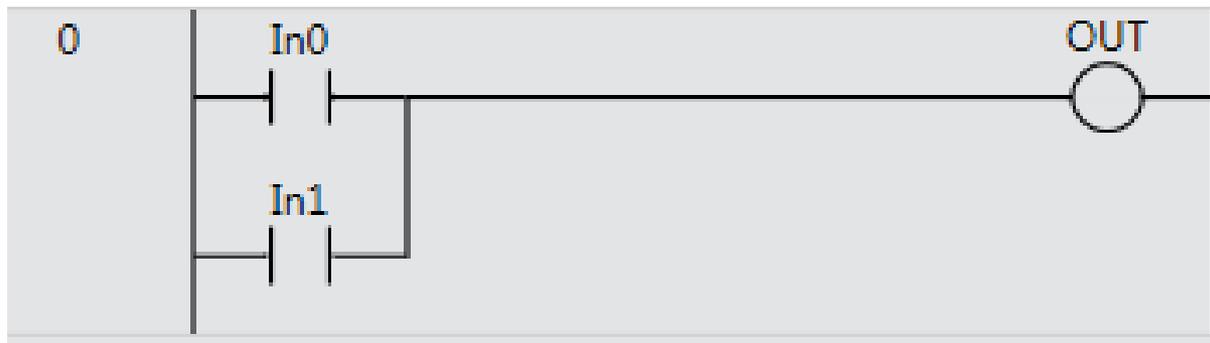


SCHEMA ELETTRICO

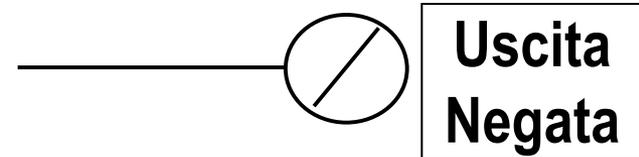
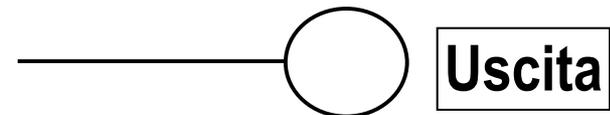
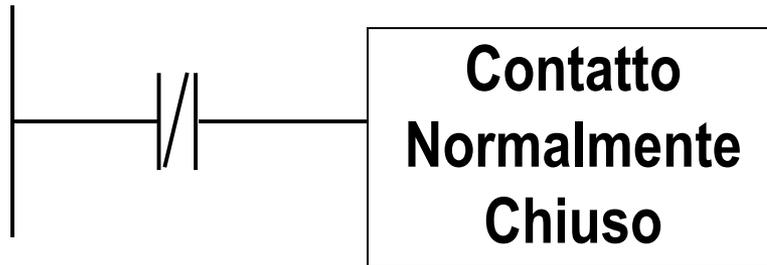
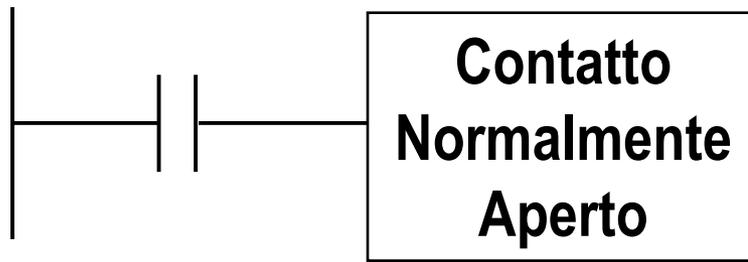


IN0	IN1	OUT
0	0	0
0	1	1
1	0	1
1	1	1

- Esempio



Simboli



Contatti

Ci sono tre modi per inserire un contatto:

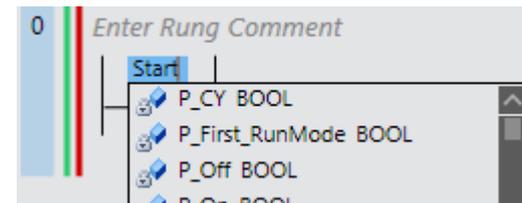
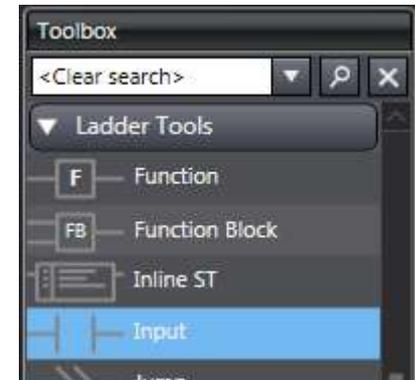
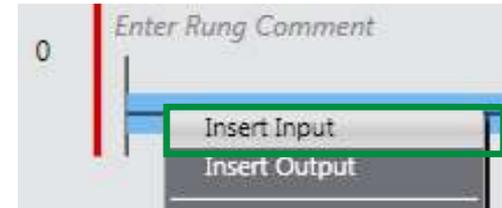
1. Tasto destro sulla rung e clic su **“Inserisci ingresso”**
2. Posizionare il cursore sulla rung e premere il tasto **“C”**;
3. Trascinare un elemento **Ingresso** dagli **Strumenti ladder** della **Casella degli strumenti**

Inserito il contatto non resta che associarlo ad una variabile **BOOL**;

Durante la digitazione compare una lista di autocompletamento con i simboli **BOOL** già dichiarati.

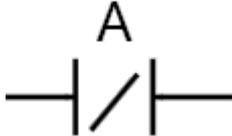
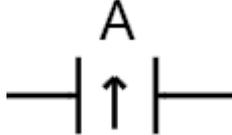
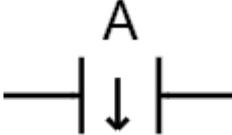
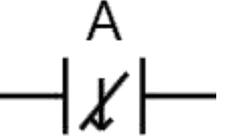
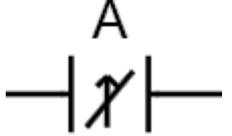
- Se il nome digitato non corrisponde a nessuna variabile esistente viene creata una nuova variabile **Interna**.

Nota: Se il nome digitato corrisponde ad una delle **variabili globali**, essa verrà aggiunta nella scheda **Esterne**.



First_Section		
	Name	Data
Internals	Start	BOOL
Externals		

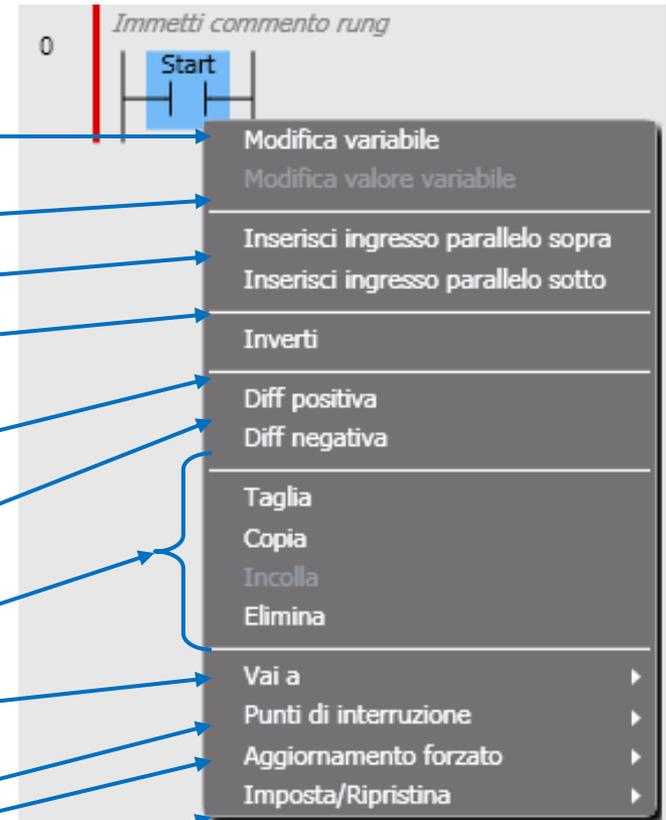
Varianti del contatto:

- Contatto negato: 
- Contatto differenziato:  
- Contatto negato e differenziato:  

Contatti

Cliccando col tasto destro è possibile selezionare le seguenti opzioni:

- Cambia la variabile associata al contatto;
- Aggiungi un contatto in parallelo sopra;
- Aggiungi un contatto in parallelo sotto;
- Rendi un contatto negato;
- Chiudi il contatto solo sul fronte di salita;
- Chiudi il contatto solo sul fronte di discesa;
- Taglia / Copia / Incolla/ Elimina il contatto;
- Naviga attraverso il programma;
- Imposta un breakpoint per la simulazione;
- Forza la variabile a ON o a OFF;
- Imposta la variabile a ON o a OFF.

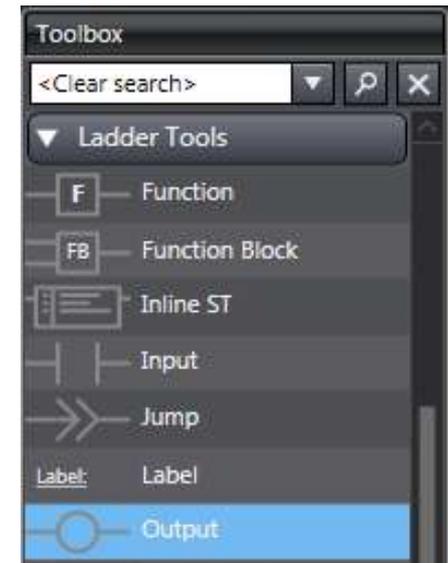


Bobine

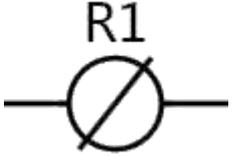
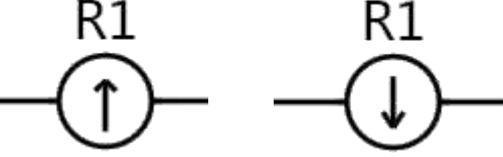
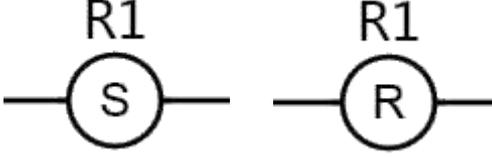
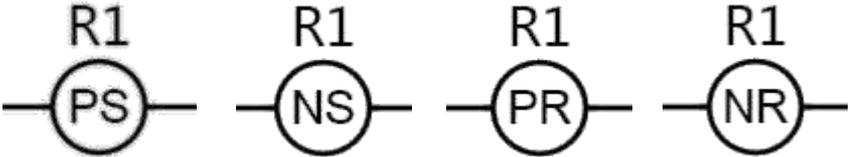
Ci sono tre modi per inserire una bobina:

1. Tasto destro sulla rung e scegliere ***“Inserisci Uscita”***
2. Posizionare il cursore sulla rung e premere il tasto ***“O”***;
3. Trascinare l’elemento ***Uscita*** dagli ***Strumenti Ladder*** dalla ***Casella degli strumenti***

Quindi associare una variabile (stessa procedura del contatto).



Variante della bobina:

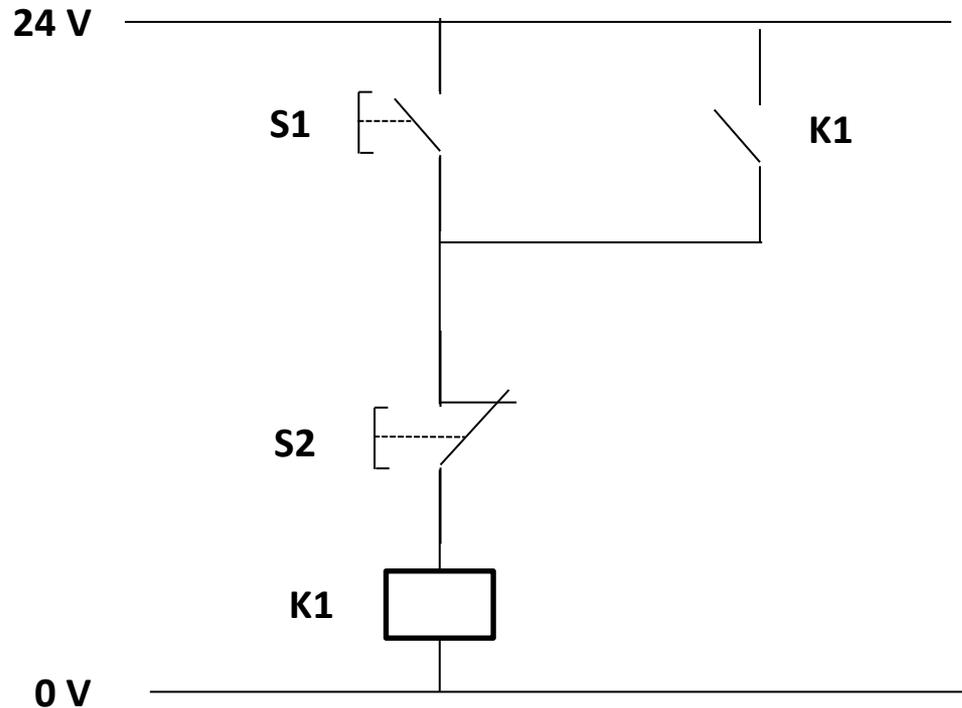
- Bobina negata: 
- Bobina differenziata: 
- SET e RESET: 
- SET e RESET differenziati: 

Bobine

Cliccando col tasto destro è possibile selezionare le seguenti opzioni:

- Cambia la variabile associata alla bobina;
- Aggiungi una bobina in parallelo sopra;
- Aggiungi una bobina in parallelo sotto;
- Rendi la bobina negata;
- Attiva la bobina sul fronte di salita della rete che la precede;
- Attiva la bobina sul fronte di discesa della rete che la precede;
- Istruzioni SET o RESET;
- Taglia / Copia / Incolla/ Elimina la bobina;
- Naviga attraverso il programma;
- Imposta un breakpoint per la simulazione;
- Forza la variabile a ON o a OFF;
- Imposta la variabile a ON o a OFF





Esercizio: Circuito di autoritenuta

Soluzione

Sezione0 - Programma0 X

Variabili

Namespace - utilizzato

Interne	Nome	Tipo dati	Valore iniziale
Esterne	S1	BOOL	
	S2	BOOL	
	K1	BOOL	

0

```

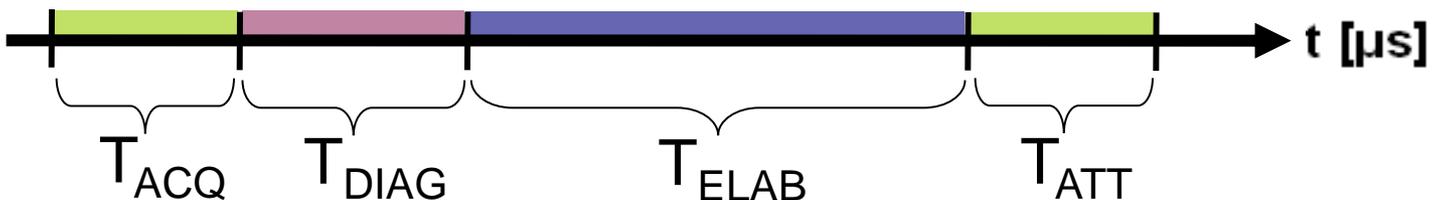
graph LR
    S1_1[S1] --- K1_1((K1))
    S1_2[S1] --- K1_2((K1))
    K1_1 --- OR(( ))
    K1_2 --- OR
    OR --- S2[S2]
    S2 --- K1_3((K1))
  
```

Il tempo di scansione

Il tempo di scansione è il tempo che intercorre tra un I/O refresh ed il successivo

E' composto dalla somma di:

- tempo di elaborazione
- tempo di I/O refresh
- tempo di gestione dei processi comuni



Minimo segnale rilevabile

Il segnale elettrico interpretato dall'interfaccia di I/O del MAC viene letto ad ogni ciclo del Task Primario

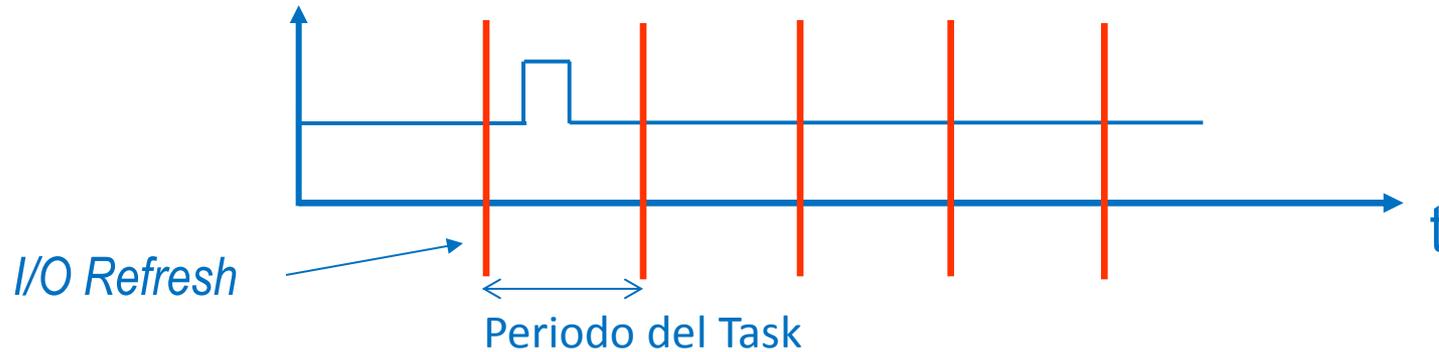
Ciò implica che se un segnale ha una variazione più breve del tempo di ciclo potrebbe non essere rilevato

Per avere la certezza che un segnale fisico sia sicuramente rilevato dal MAC è necessario quindi che la sua durata sia superiore al tempo di ciclo

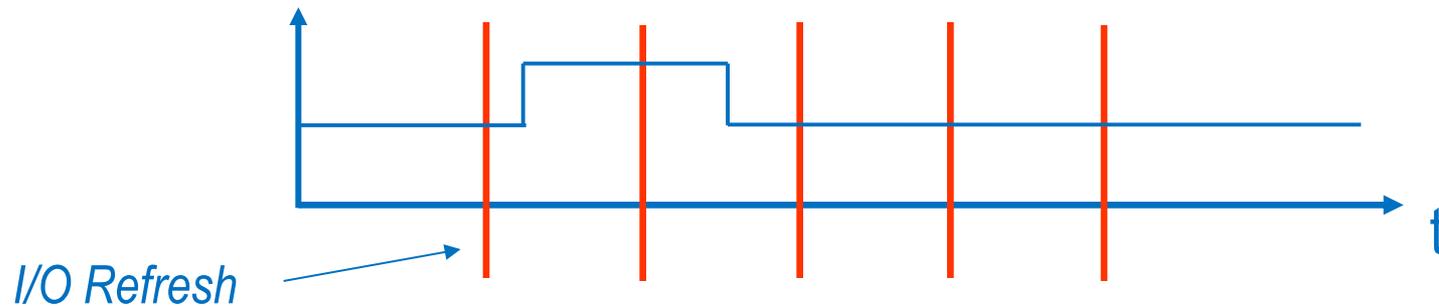


Minimo segnale rilevabile

- Segnale di ingresso non sempre rilevabile:

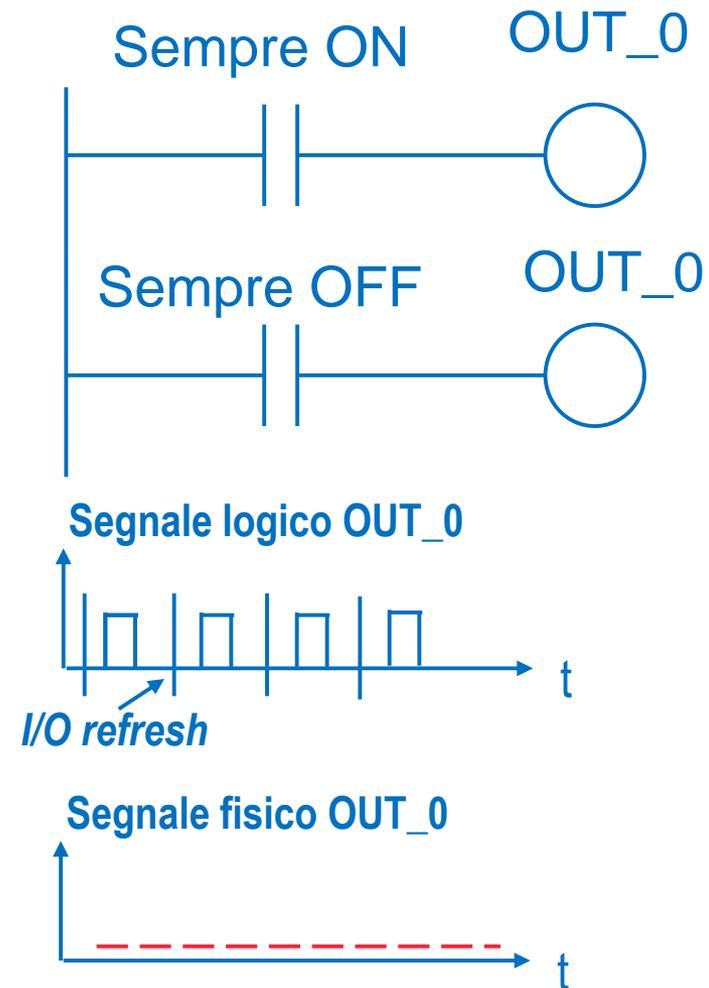


- Segnale di ingresso sicuramente rilevabile:



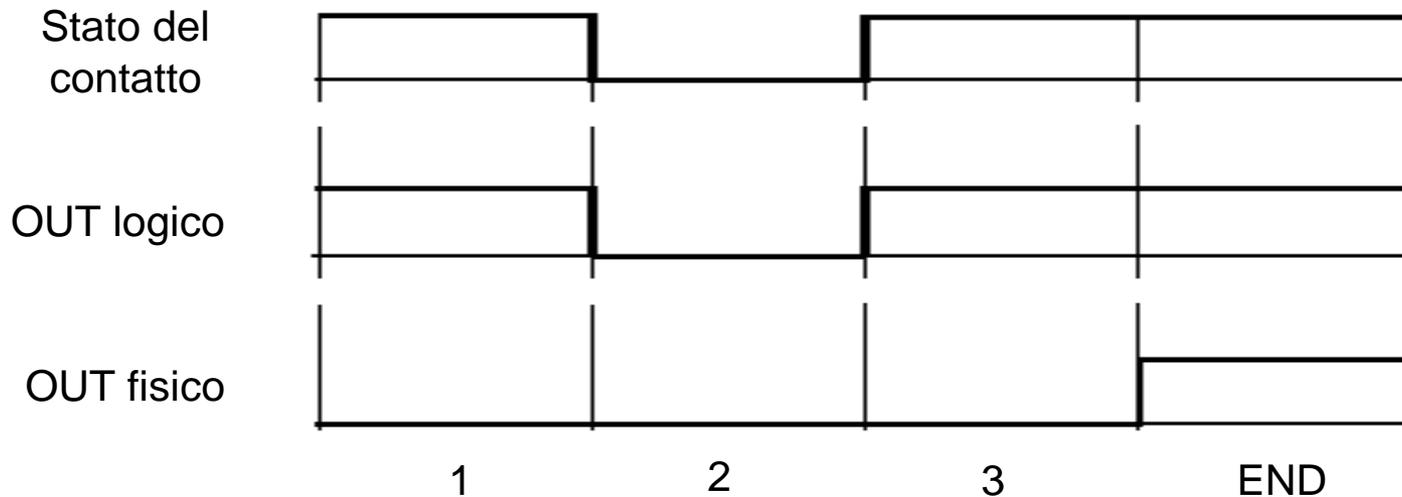
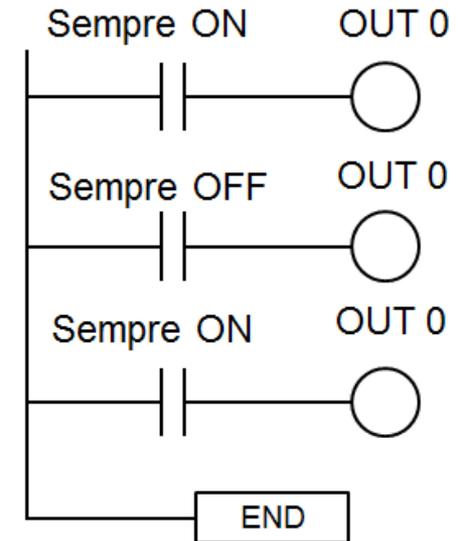
Stato logico e stato fisico

- La modifica dello stato logico di un'uscita durante l'elaborazione del programma **non implica** la modifica del suo stato fisico
- Lo stato fisico è determinato solo dall'ultima modifica dello stato logico presente nell'Immagine di Processo



Stato logico e stato fisico

N. rung	Stato del contatto	OUT logico	OUT fisico
1	ON	ON	OFF
2	OFF	OFF	OFF
3	ON	ON	OFF
END		ON	ON





DOMANDE?



Sistemi di codifica

Corso Base Machine Controller

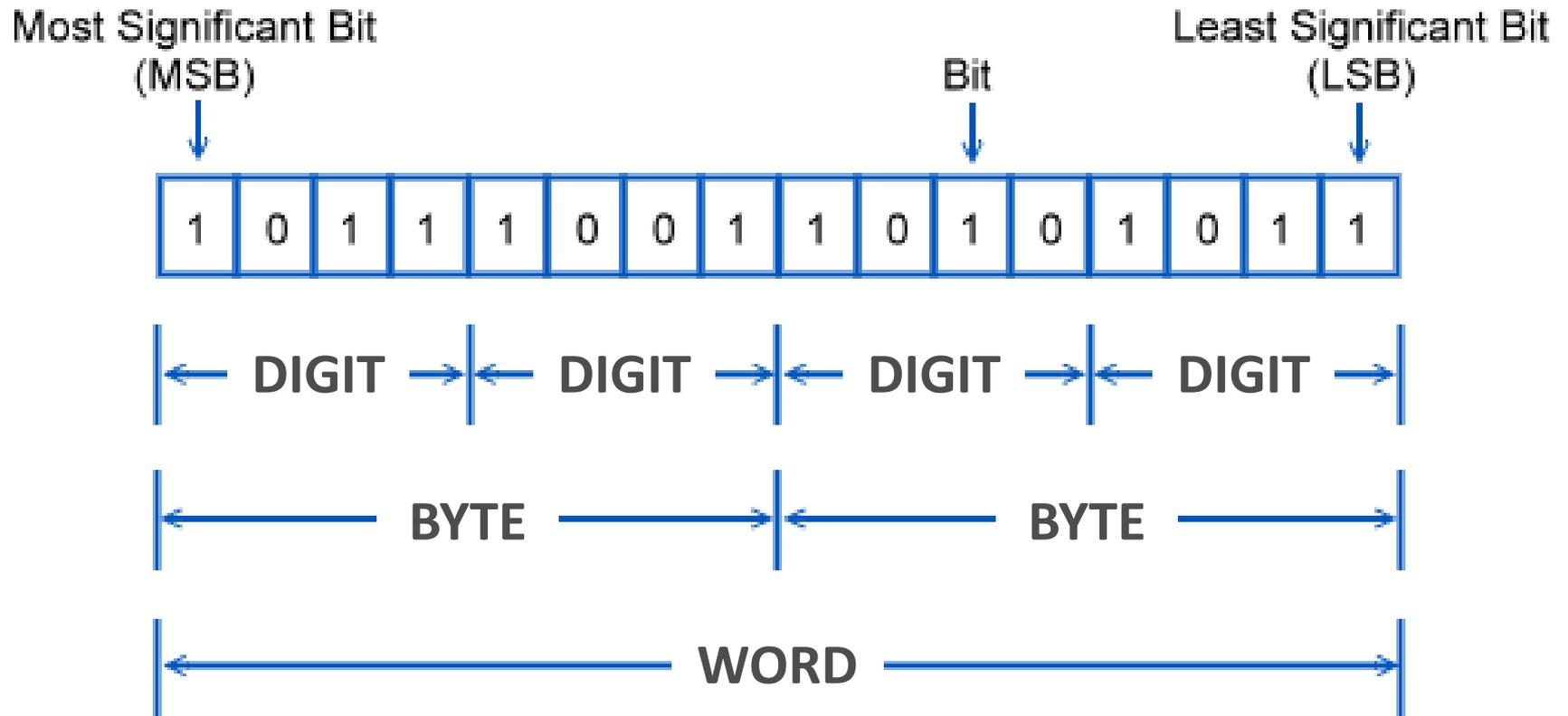
Bit

L'elemento di memoria elementare si chiama bit

«Un bit è l'unità di misura dell'informazione (dall'inglese "**b**inary **d**igit"), definita come la quantità minima di informazione che serve a discernere tra due possibili eventi equiprobabili.»



Le informazioni: Bit, Byte e Word



Il valore di un numero e le sue rappresentazioni

- Un numero è composto da una serie di cifre che rappresenta un valore
- Nella codifica decimale esistono 10 cifre che rappresentano un valore in funzione della loro posizione
- Ad esempio il valore del numero 1234 si calcola come:
- $1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0 = 1234$
- Il fattore moltiplicativo si compone di una base (nel sistema decimale 10) e di un esponente che varia da 0 a n a seconda della posizione della cifra
- Le cifre più a destra sono quindi cifre meno significative

La codifica in base n

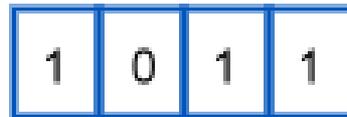
- Per convertire un numero di n cifre in una qualunque base B in un valore decimale si deve operare come segue

$$N_{\text{dec}} = B_{n-1} \times 1^{\text{cifra}} + B_{n-2} \times 2^{\text{cifra}} + \dots + B_0 \times n^{\text{cifra}}$$

- Le cifre a sinistra sono indicate come le cifre “più significative”

La codifica esadecimale (binaria)

- L'utilizzo di 4 bit (DIGIT) permette di rappresentare 16 stati differenti.



- La codifica esadecimale assegna questi 16 stati alle cifre da 0 a 9 e le lettere A, B, C, D, E, F



- E' possibile rappresentare in una word (4 DIGIT) un valore esadecimale da 0 a FFFF, che corrisponde ad un valore decimale da 0 a 65535.

Esempi di codifica dei numeri

Base 10

Codifica decimale: cifre da 0 a 9

4	3	2	1
---	---	---	---

$$4321 = 10^3 \times 4 + 10^2 \times 3 + 10^1 \times 2 + 10^0 \times 1 = 4.321 \text{ unità}$$

Base 2

Codifica binaria: cifre 0 e 1

1	0	1	1
---	---	---	---

$$1011 = 2^3 \times 1 + 2^2 \times 0 + 2^1 \times 1 + 2^0 \times 1 = 11 \text{ unità}$$

Base 16

Codifica esadecimale: cifre da 0 a 9 + A...F

1	A	3	2
---	---	---	---

$$1A32 = 16^3 \times 1 + 16^2 \times 10 + 16^1 \times 3 + 16^0 \times 2 = 6.706 \text{ unità}$$

Tabella di codifica

DECIMALE	ESADECIMALE	BINARIO
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111



DOMANDE?



IEC 61131-3

Corso Base Machine Controller

OMRON

OEE-I

Cos'è l'IEC 61131-3?

- La parte 3 dello Standard IEC 61131 definisce, per ciascuno dei linguaggi di programmazione comunemente usati:
 - I principali campi di applicazione;
 - Le regole sintattiche e semantiche;
 - Degli elementi di programmazione di base semplici ma completi;
 - I test e i mezzi tramite cui i produttori possano espandere o adattare questi elementi di programmazione ai propri controllori programmabili.
- L'edizione attuale dello standard IEC 61131-3 è la 3.0

IEC 61131 Parte 3: Cosa comprende

1. La struttura della programmazione:

- Task;
- Variabili;
- Tipi di dati;
- POU:
 - Funzioni;
 - Function block;
 - Programmi;
- Set istruzioni (unico per tutti i linguaggi)

2. Simboli, sintassi dei linguaggi:

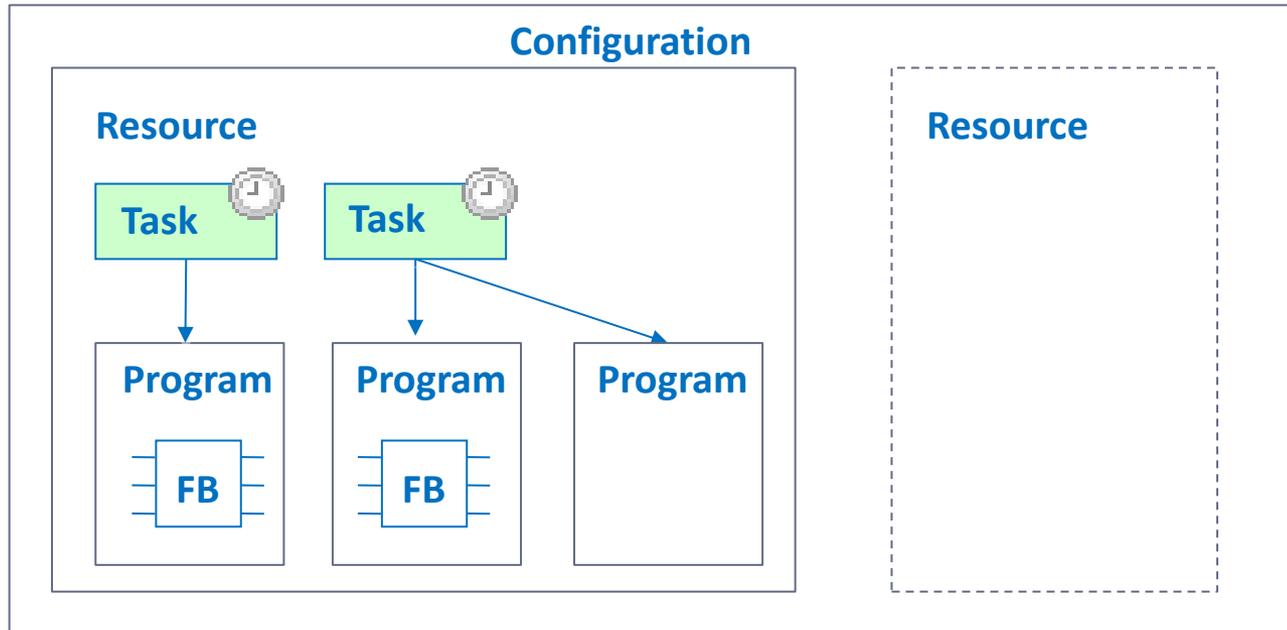
- Ladder;
- Testo Strutturato;
- Function Block diagram;
- SFC
- Lista istruzioni.



IEC 61131-3: Task

Configurazione di un sistema in IEC 61131-3

- L'immagine mostra l'organizzazione di un progetto secondo l'IEC 61131-3



Programmi, Function Block (FB) e Funzioni (FUN), sono tutti POU (Program Organization Unit).

La temporizzazione per l'esecuzione di ogni POU è definita assegnando il POU ad un task.

Cos è un task?

- Un task serve ad invocare l'esecuzione di un gruppo di programmi su base periodica, secondo un **intervallo di tempo** e un **livello di priorità** definiti a priori.
- Essi determinano pertanto la temporizzazione dell'esecuzione di uno o più programmi, ma non il contenuto.

Task, tempi e priorità

- I tipi di task disponibili nell'NJ sono i seguenti:

Tipo	Descrizione	Quantità	Priorità
Primary periodic task	<ul style="list-style-type: none"> • E' il task a priorità più alta. • E' un task periodico, per cui viene eseguito ad intervalli fissi regolari. • E' il task con il minore periodo di esecuzione, che coincide con il periodo del ciclo EtherCAT. 	1	4
Task periodico	Viene eseguito secondo un periodo fissato, negli intervalli di inattività del primary periodic task.	0-3	16, 17 o 18
Task a evento*	Viene eseguito al verificarsi di una specifica condizione	0-32	8, 48

*NJ ver. 1.03+

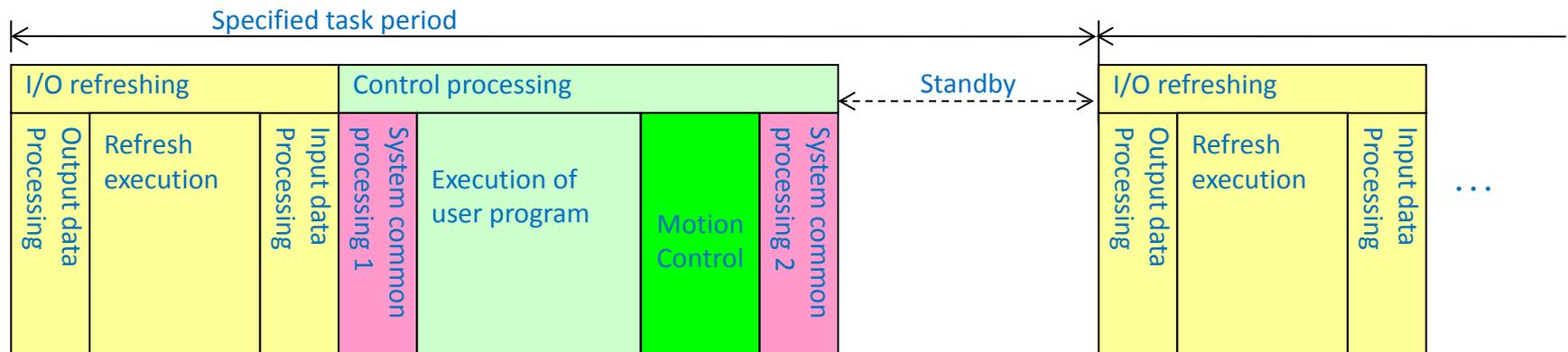
Periodo di un task

Oggetto	Specifica	
Primary periodic task	500 μ s*, 1ms, 2ms, 4ms	
Task periodici	L'intervallo di tempo assegnato ai task periodici sar� sempre un multiplo intero del tempo del primary periodic task	
	Intervallo di tempo del primary periodic task	Possibili tempi per I task periodici
	500 μ s	1ms, 2ms, 3ms, 4ms, 5ms, 8ms, 10ms, 15ms, 20ms, 25ms, 30ms, 40ms, 50ms, 60ms, 75ms, 100ms
	1ms	1ms, 2ms, 3ms, 4ms, 5ms, 8ms, 10ms, 15ms, 20ms, 25ms, 30ms, 40ms, 50ms, 60ms, 75ms, 100ms
	2ms	2ms, 4ms, 8ms, 10ms, 20ms, 30ms, 40ms, 50ms, 60ms, 100ms
4ms	4ms, 8ms, 20ms, 40ms, 60ms, 100ms	

*: NJ301 ver 1.03+, NJ501 1.00+

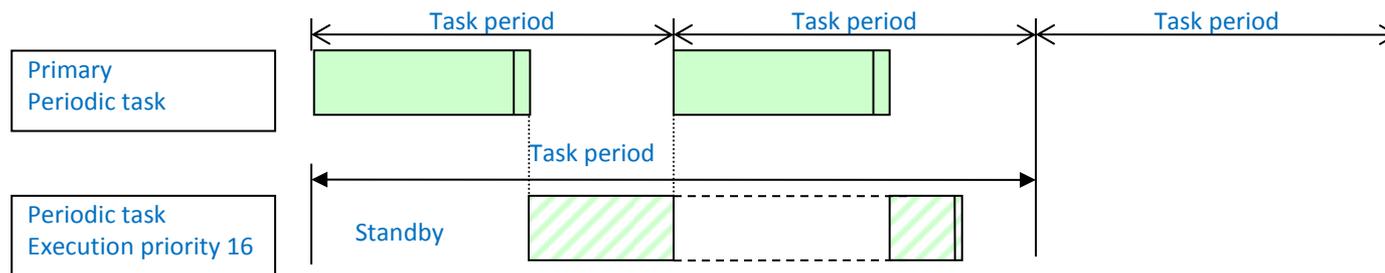
Primary periodic task

- E' il task con priorità di esecuzione più alta;
- Dev'essere dichiarato uno e un solo primary periodic task.
- All'interno di un singolo periodo del primary periodic task vengono eseguite le operazioni di:
 - I/O refresh dei segnali delle schede del rack e degli slave EtherCAT
 - Elaborazione dei processi comuni,
 - Esecuzione del programma utente
 - Elaborazione delle funzioni di motion control.



Task periodici

- Il task periodico abilita l'esecuzione dei programmi ad esso assegnati ad intervalli di tempo prefissati, multipli interi del periodo del primary periodic task.
- Ai task periodici vengono assegnati programmi che non richiedono la stretta sincronia con il ciclo EtherCAT.
- Possono essere dichiarati fino a 3 task periodici.
- I possibili livelli di priorità sono 16, 17 e 18.
- L'I/O refresh dei moduli del rack e degli slave EtherCAT può essere assegnato soltanto al task periodico a priorità 16 e al primary periodic task.





IEC 61131-3: POU (Program Organization Unit)

POU: Cos'è un POU?

- Un POU è un'unità elementare del programma utente. Nello standard IEC61131-3 è definito semplicemente:

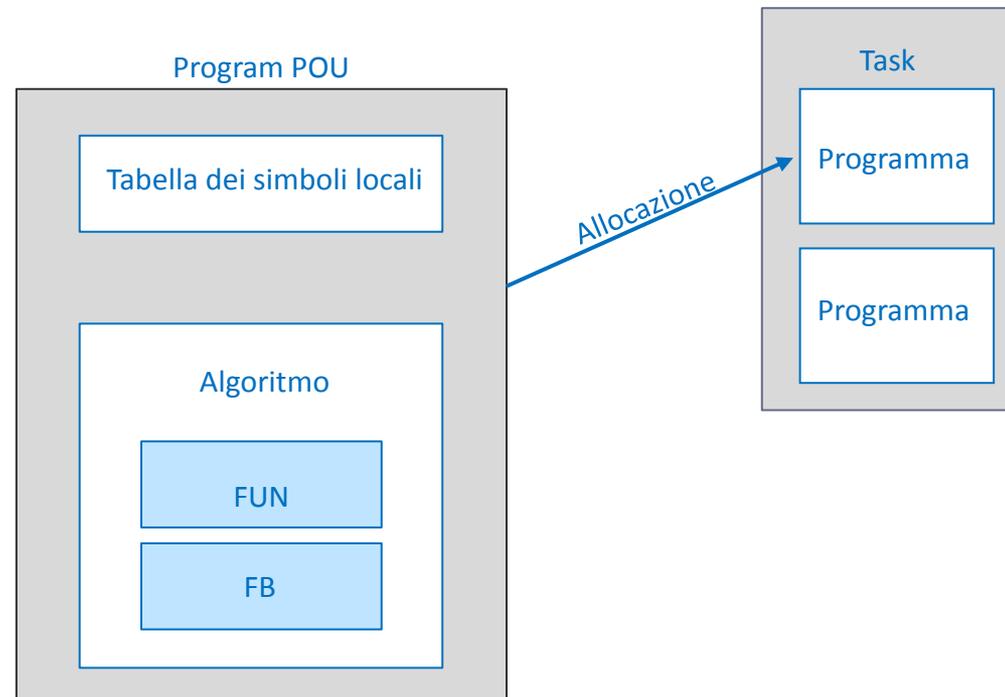
«Program Organization Unit: Una funzione, function block o programma.»

- Un programma utente si compone sempre da uno o più POU.
- **I tipi di POU sono tre:**
 - Funzioni (FUN)
 - Function Block (FB)
 - Programmi
- Tutti i POU consistono in un algoritmo e una tabella dei simboli locali



POU: Programmi

- L'associazione ad un task determina le tempistiche di esecuzione del programma



Un POU di tipo programma non può essere invocato da un altro POU

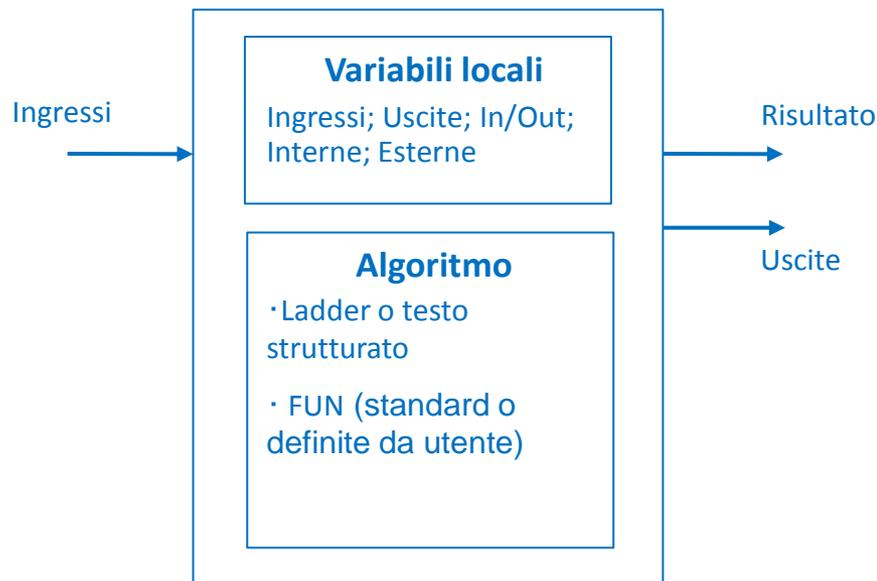
POU: Funzioni (FUN)

- Una FUN è un blocco di codice riutilizzabile;
- Una FUN è un POU il quale, quando eseguito, restituisce un solo valore (il **risultato**).
- Arbitrariamente, possono essere fornite delle uscite supplementari.
- La condizione per l'esecuzione è subordinata allo stato dell'ingresso *EN*.

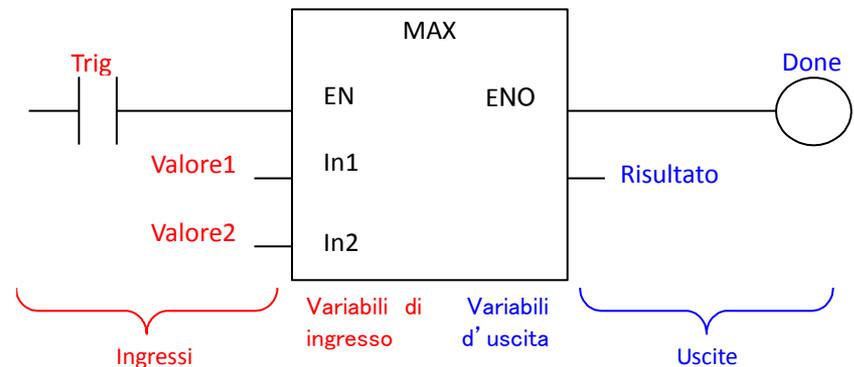
POU: Funzioni (FUN)

- Quando utilizzati nel programma utente viene invocato il nome della funzione

Struttura FUN



Esempio Ladder



Esempio testo strutturato

```
Risultato := MAX(In1:=Valore1, In2:= Valore2);
```

Oppure:

```
Risultato := MAX( Valore1, Valore2 );
```

Oppure è possibile assegnare anche EN/ENO:

```
Risultato := MAX( EN:=Trig, In1:= Valore1, In2:= Valore2, ENO=>Done);
```

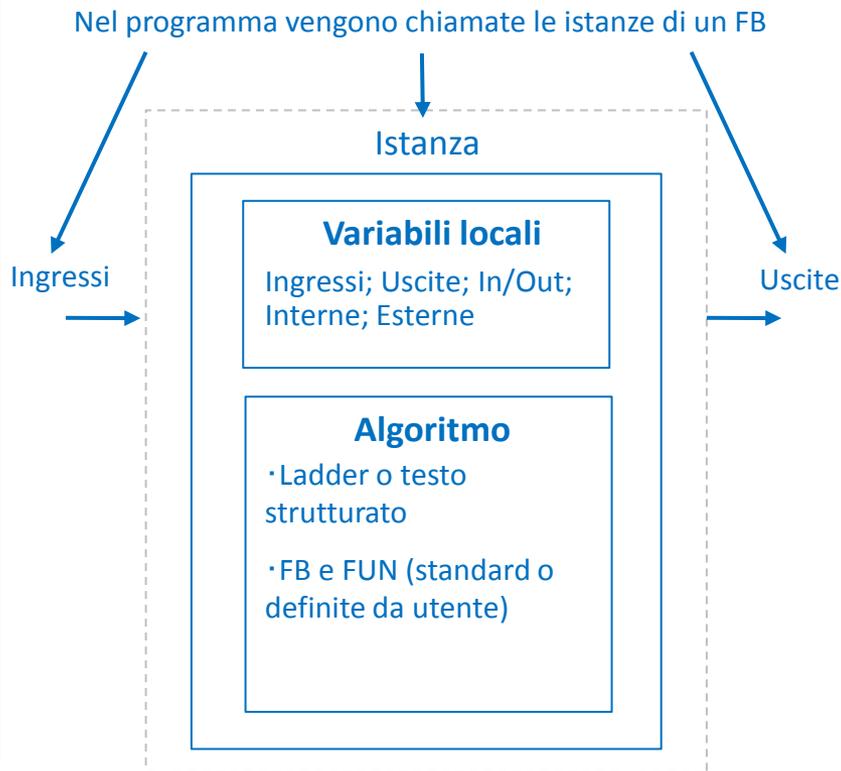
POU: Function Block (FB)

- Un FB è un blocco di codice riutilizzabile;
- Un FB è un POU che, quando eseguito, restituisce uno o più valori.
- Le copie di un FB sono chiamate «istanze».
- Ciascuna istanza mantiene memoria dello stato delle variabili interne, tra un'esecuzione e la successiva, per cui l'invocazione dello stesso FB con gli stessi ingressi può dare risultati diversi.

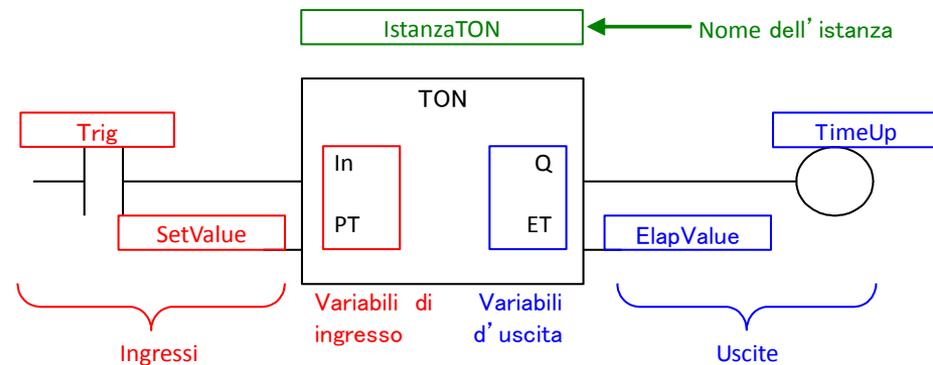
POU: Function Block (FB)

- Quando usati nel programma utente viene invocata un'istanza del FB

Struttura FB



Esempio Ladder



Esempio testo strutturato

```
IstanzaTON(In:=Trig, PT:=SetValue, Q=>TimeUp, ET
=>ElapValue);
```

Oppure

```
IstanzaTON(Trig, SetValue, TimeUp, ElapTime);
```

*Nel secondo caso devo assegnare tutti gli input e gli output, nel primo caso posso anche ometterne qualcuno non necessario

Rosso: Ingressi

Blu: Uscite

POU: Funzioni vs Function Block

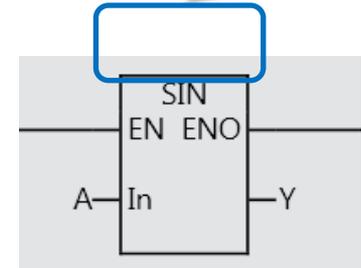
Funzioni (FUN):

- Non necessitano di istanza:
 - Sono più semplici da utilizzare;
 - Non occupano memoria dati
 - Possono essere utilizzati illimitatamente
- Necessitano di un ingresso EN
- Hanno sempre una variabile risultato

Function Block (FB)

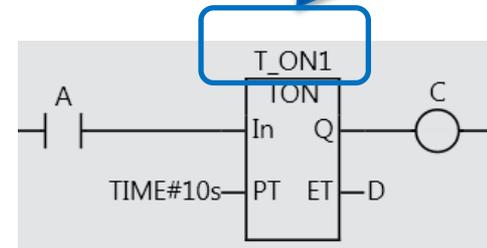
- Necessitano di istanza per poter essere utilizzate nel programma:
 - Occupano memoria dati
 - Il numero di istanze è limitato dalla memoria dati disponibile
- Si usano quando è necessario memorizzare valori tra un ciclo e l'altro, o quando contengono FB nel codice (ad esempio timer o FB utente)

Non serve istanziare



```
1 Y:=SIN(A);
2
```

Necessita di un'istanza



```
1 T_ON1(A, TIME#10s, C, D);
2
```

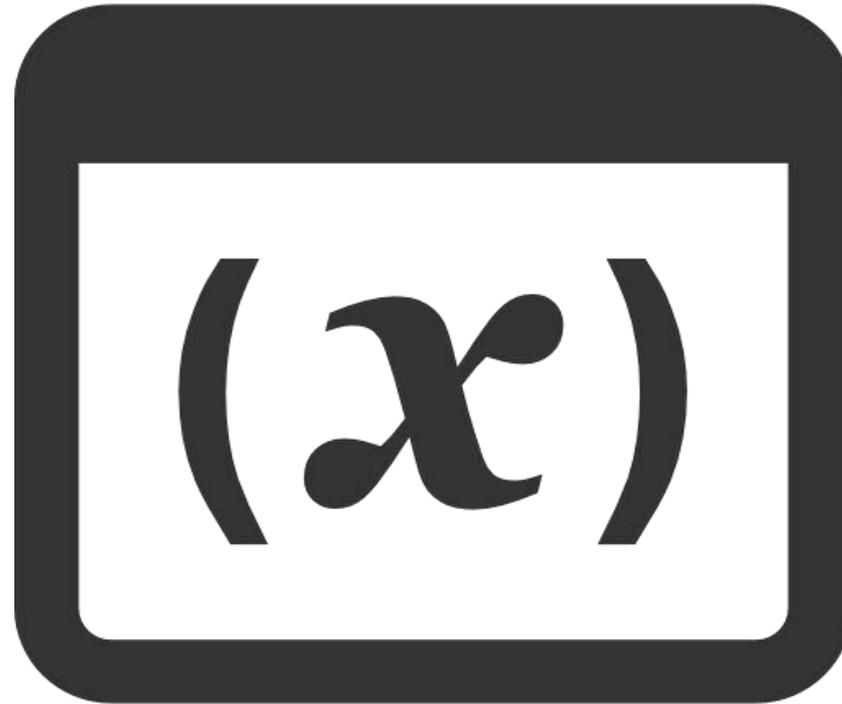
POU: Caratteristiche

Oggetto		Programma	Function Block (FB)	Funzione (FUN)
Associazione a task		Possibile	Non possibile	Non possibile
Variabili locali		Il valore è mantenuto	Il valore è mantenuto. Ogni istanza ha variabili locali.	Il valore non è mantenuto.
Variabile risultato		Nessuna	Nessuna	Si
Algoritmo interno	Linguaggi	LD, ST	LD, ST	LD, ST
	Può contenere FUN	Si*1	Si*1	Si*1
	Può contenere FB	Si*1	Si*1	No
Istanza		No	Si*2	No
Condizione per l'esecuzione		Sempre eseguito	Sempre eseguito *3	Attivazione dell'ingresso EN
EN/ENO		No	EN: No*4 ENO: Si (facoltativo)	EN: Si. ENO: Si (facoltativo)

NOTE:

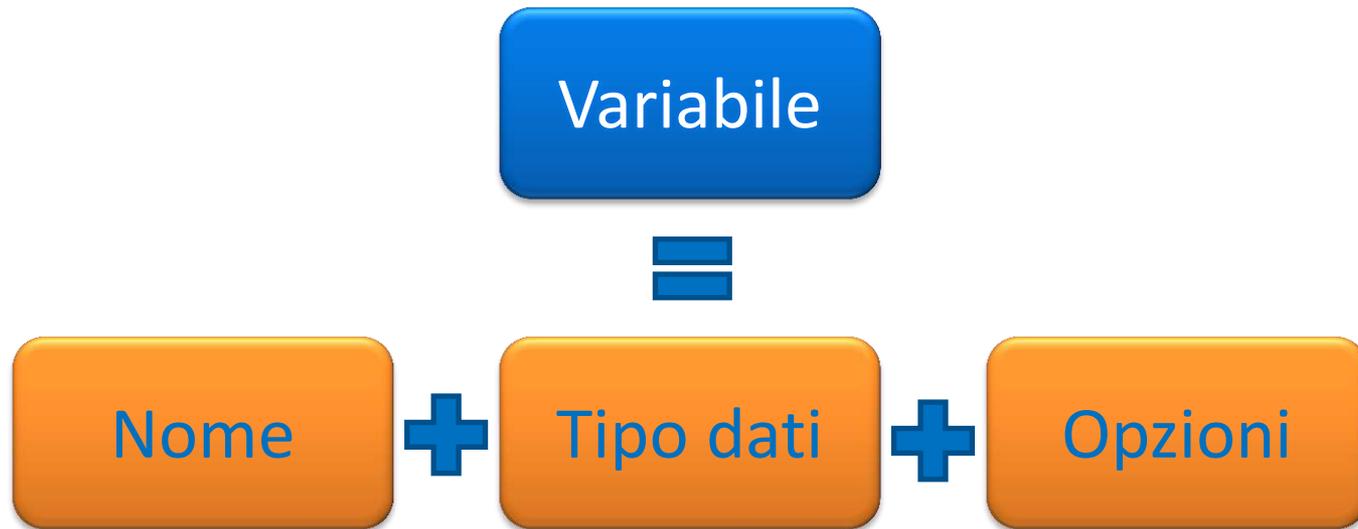
1. Sia i FB che le FUN possono essere innestati uno dentro l'altro, fino ad un massimo di 8 livelli.
2. Le istanze dei FB sono allocate sempre e solo come variabili locali
3. L'eventuale ingresso booleano non serve per abilitare il FB, a meno che non sia implementato da programma.

Per usare il FB all'interno di un programma ladder, sono necessari almeno un ingresso e almeno un'uscita booleana.



IEC 61131-3: Variabili

Caratteristiche delle variabili



- La programmazione del Sysmac NJ è basata esclusivamente su **variabili** e non su indirizzi.
- Una variabile è definita con un **nome**, un **tipo dati** e delle **opzioni**.
- Una variabile può essere **locale** oppure **globale**.

Variabili: variabili locali

- Ogni POU ha una tabella in cui vengono dichiarate tutte variabili locali che possono essere utilizzate al suo interno.
- In genere le variabili dichiarate in un POU sono accessibili solo da quel POU (variabili locali)

Variabili: variabili locali

Internals	Name	Data Type	Initial Value	AT	Retain	Constant	Comment
Externals	A	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	
	B	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	
	C	BOOL			<input type="checkbox"/>	<input type="checkbox"/>	

```

1 A:=SIN(2.0*B);
2 C:=MiaFUN(1,1,1,1);

```

Variabili: variabili locali

- A seconda del tipo di POU (programma, FUN o FB) c'è una varietà di variabili locali utilizzabili:

Programma	FUN	FB
Interne	Interne	Interne
Esterne	Input	Input
	Output	Output
	In/Out	In/Out
	Risultato	Esterne
	Esterne	

Variabili: variabili globali

- Sono variabili comuni a tutto il sistema.
- Possono essere variabili collegate allo hardware (es. variabile booleana associata ad un ingresso digitale);
- Possono essere accessibili da tutti i POU;
- Possono essere pubblicate come tag in una rete Ethernet/IP.

Variabili: opzioni di una variabile

Opzione	Descrizione
Nome	Nome della variabile (set caratteri ISO-646 IRV , no keyword, no spazi, no doppi underscore)
Tipo dati	Data type della variabile. Può essere standard oppure derivato
Valore iniziale	Impostazione di un valore specifico nelle seguenti situazioni: <ul style="list-style-type: none"> • All'accensione dell'NJ • Quando l'NJ viene portato in modalità RUN
AT	Associazione della variabile ad un percorso hardware (es. PDO di slave EtherCAT, variabile di un modulo locale) o emulazione indirizzo CJ (es. DM1000, W123.1)
Mantieni	Ritentività del dato allo spegnimento dell'NJ
Costante	Inibisce la variazione del dato.
Pubblicazione di rete (solo globali)	Rende la variabile accessibile esternamente tramite comunicazione con messaggi CIP o funzione data link in Ethernet/IP. E' possibile specificare "Pubblica solo", "Ingresso", "Uscita" o "Non pubblicare"
Commento	Consente di specificare un commento descrittivo per la variabile
Fronte (solo ingressi FB)	Attiva la variabile solo sui fronti (di salita o di discesa) dell'ingresso del FB

Variabili: Assegnazione di valori

- L'espressione di valori costanti nel programma utente richiede una impostazione del valore con il relativo tipo dati, separati da un #.

[TIPO DATI]#[VALORE]

Variabili: Assegnazione di valori

- Esempio: Espressione del valore numerico 18 tramite il tipo INT.

INT#18

- E' possibile specificare la base in cui il numero è espresso, effettuando un secondo cast dopo il tipo dati:
 - *INT#10#18* (codifica decimale)
 - *INT#16#0012* (codifica esadecimale)
 - *INT#8#00000022* (codifica ottale)
 - *INT#2#00000000000010010* (codifica binaria)
- Si possono inserire degli underscore (`_`) che non influiscono sul valore della costante, es. *INT#2#0000_0000_0001_0010*

NOTA: Il cast è utilizzabile per tutti i tipi di dati ad esclusione delle stringhe di testo

Emulare indirizzi CJ

- L'opzione AT consente di dichiarare un percorso a cui una variabile punta.
- Questo percorso può essere il percorso di un dispositivo collegato (uno slave in EtherCAT o un modulo CJ sul rack), oppure può essere usato per emulare indirizzi propri dell'architettura di memoria CJ.
- La sintassi è:

IOM://[indirizzo] oppure %[indirizzo]

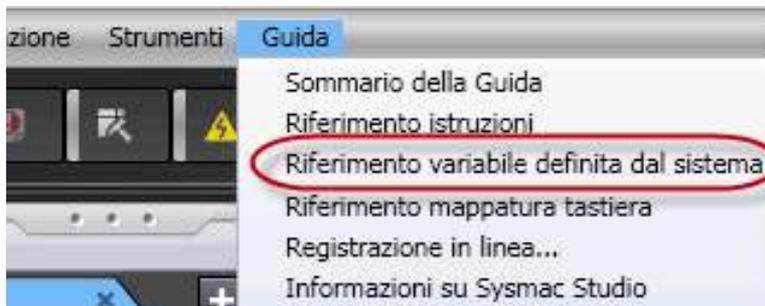
Ad esempio: IOM://6143.15

IOM://W511

IOM://D1000

Variabili di sistema

- Sono variabili predefinite, usate per monitorare lo stato di determinate funzionalità del dispositivo (rete Ethernet/IP, EtherCAT, ...)
- Queste variabili sono anche preregistrate nella tabella delle variabili globali (sebbene non visibili).
- Il loro nome inizia sempre per «_» (underscore) o per «P_».



Variabili di sistema

- Nel nome delle variabili di sistema il prefisso indica a quale campo funzionale esse fanno riferimento :

Modulo funzionale	Prefisso (categoria)
Modulo PLC	_PLC
	_CJB
	_ (senza categoria *1)
Modulo Motion Control	_MC
Modulo EtherNet/IP	_EIP
Modulo EtherCAT master	_EC

*1 Alcune variabili nel campo funzionale PLC non sono incluse nelle categorie _PLC o _CJB, ma sono legate al funzionamento generale della CPU, come SD card, orologio, tasks, errori, alimentazione, comunicazione, programmazione (variabili «P_»)



IEC 61131-3: Tipi di dati

Tipi di dati base: Caratteristiche

Classificazione	Tipo dati	Dimensione (Byte)	Intervallo di valori	Esempio di assegnazione di un valore costante
BOOL	BOOL	2	0/1 oppure FALSE/TRUE	TRUE FALSE BOOL#1
Stringhe di bit	BYTE	1	16#00 ÷ FF	BYTE#16#A5 BYTE#2#10100101 BYTE#2#1010_0101
	WORD	2	16#0000 ÷ FFFF	Nota: L'espressione del dato in base 10 non è consentita
	DWORD	4	16#00000000 ÷ FFFFFFFF	
	LWORD	8	16#0000000000000000 ÷ FFFFFFFFFFFFFFFF	
Interi	SINT	1	-128 ÷ +127	
	INT	2	-32768 ÷ +32767	Se non è specificato nessun cast il dato viene interpretato come DINT
	DINT	4	-2147483648 ÷ +2147483647	
	LINT	8	-9223372036854775808 ÷ +9223372036854775807	
	USINT	1	0 ÷ +255	
	UINT	2	0 ÷ +65535	
	UDINT	4	0 ÷ +4294967295	
	ULINT	8	0 ÷ +18446744073709551615	

Tipi di dati base: Caratteristiche

Classificazione	Tipo dati	Dimensione (Byte)	Intervallo di valori	Esempio di assegnazione di un valore costante
Virgola mobile	REAL	4	Cifre significative: 7 cifre -3.402823e+38 ÷ -1.175494e-38 0 -1.175494e-38 ÷ 3.402823e+38 +∞/-∞	REAL#10#-12.0 Se non è specificato nessun cast il dato viene interpretato come LREAL
	LREAL	8	Cifre significative: 15 cifre -1.79769313486231e+308 ÷ -2.22507385850720e-308 0 2.22507385850720e-308 ÷ 1.79769313486231e+308 +∞/-∞	
Durata	TIME	8	T#-9223372036854.775808ms ÷ T#+9223372036854.775807ms	TIME#60s500ms , T#60s500ms T#16d5h3m4s
Data	DATE	8	D#1970-01-01 ÷ D#2554-07-21	DATE#2010-12-3 ,D#1994-09-23
Ora	TIME_OF_DAY	8	TOD#00:00:00.000000000 ÷ TOD#23:59:59.999999999	TOD#12:16:28.12
Data e ora	DATE_AND_TIME	8	DT#1970-01-01-00:00:00.000000000 ÷ DT#2554-07-21-23:34:33.709551615	DT#1994-09-23-12:16:28.12
Stringa	STRING	0 - 1985	Codifica UTF-8.	'OMRON' 'PLC'

Tipi di dati base: Prefissi e loro significati

- U[] [] []: Unsigned
- S[] [] []: 1 byte
- D[] [] []: 4 byte
- L[] [] []: 8 byte

Es.: DWORD → Esadecimale a 32 bit

USINT → Intero senza segno a 8 bit

LREAL → virgola mobile a 64 bit

Tipi di dati derivati

- Sono tipi di dati basati sui tipi elementari e possono essere usati in aggiunta a questi.
- I tipi di dati derivati definiti nello standard IEC61131-3 sono:
 - Enumerazioni
 - Subrange
 - Strutture
 - Array
- Estensioni dello standard:
 - Unioni

NOTA: Nel Sysmac Studio, Subrange e array vengono specificati all'atto della dichiarazione di una variabile, direttamente nel campo «Tipi di dati», senza perciò definire un nuovo tipo dati.

Enumerazioni, strutture e unioni devono invece essere definite in una tabella dedicata ai nuovi tipi di dati.

Tipi di dati derivati: Enumerazioni

- Una enumerazione è una serie di «etichette» (enumeratori) associabili a dei valori numerici interi.
- Lo scopo è l'uso delle etichette di testo nel programma utente piuttosto che valori numerici, in modo da interpretarne immediatamente e intuitivamente il significato.

• Esempio:

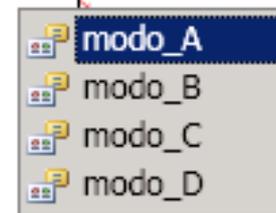
	Nome	Valore enum
▼	Modalita_avvio	
	modo_A	10
	modo_B	65
	modo_C	42
	modo_D	156



Nome	Tipo dati
MiaVar	Modalita_avvio

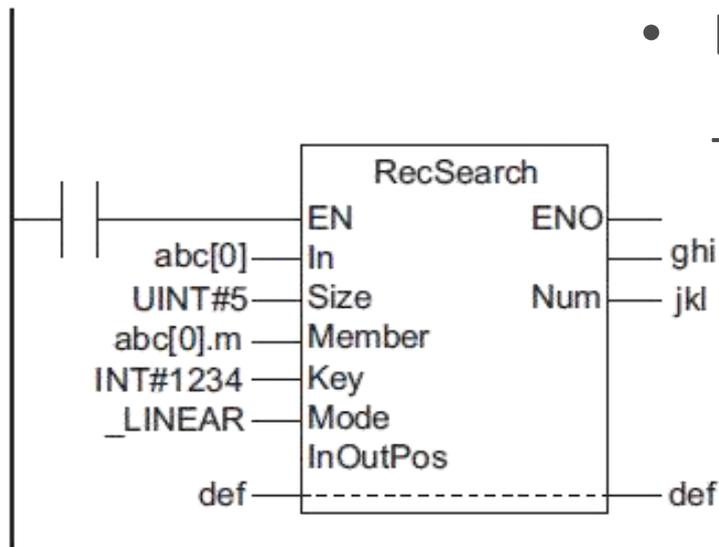
• Nel programma utente:

```
5 MiaVar:=Modalita_avvio#
```



Tipi di dati derivati: Enumerazioni

- Il cast non è necessario se la variabile enumerata è un ingresso di un FB o FUN:



- L'ingresso «Mode» è di tipo `_eSEARCH_MODE`, così composto:

Nome	Valore enum
<code>_eSEARCH_MODE</code>	-
<code>_LINEAR</code>	Lineare
<code>_BIN_ASC</code>	Ordine binario crescente
<code>_BIN_DESC</code>	Ordine binario decrescente

Tipi di dati derivati: Enumerazioni

Alcune regole:

- Le etichette (enumeratori) possono essere composte da soli caratteri alfanumerici. Non sono distinte le maiuscole dalle minuscole. Non sono consentiti caratteri speciali o parole riservate.
- Non si possono dichiarare variabili con lo stesso nome di un enumeratore.
- Il valore numerico associato a ciascun enumeratore deve essere intero, con segno, compreso tra -2147483648 e +2147483647.
- Il valore associato ad un enumeratore non può essere usato per effettuare comparazioni.
- Il numero massimo di enumeratori per una enumerazione è 2048. (il minimo è 1).
- I messaggi CIP accedono ai valori numerici anziché ai nomi degli enumeratori.

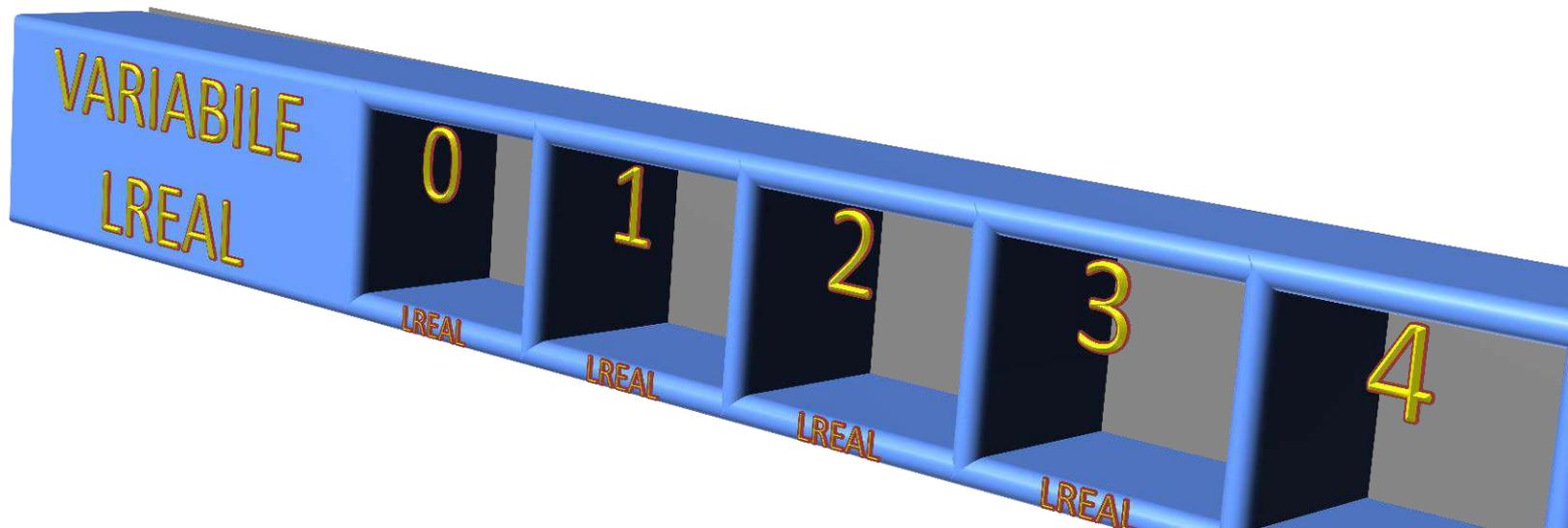
Tipi di dati derivati: Sub-range

- Un sub-range specifica l'intervallo di valori che può assumere una variabile;
- Una variabile in cui si dichiara un sub-range può assumere solamente valori compresi nell'intervallo dichiarato, inclusi i limiti inferiore e superiore;
- La specificazione di un sub-range si può applicare soltanto a **tipi numerali interi**: SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, o a tipi derivati da questi
- La sintassi per la specificazione di un sub-range è:
 - *[TIPO DATI INTERO] ([Limite inferiore]..[Limite superiore])*

Ad esempio: *INT (-4196..+4095)*

Tipi di dati derivati: Array

- E' possibile definire un "vettore" di elementi, tutti dello stesso tipo, la cui posizione è discriminata da un indice.
- La sintassi per la specificazione di una variabile array è:
 - **ARRAY**[*[Primo elemento]..[Ultimo elemento]*] **OF** [*Tipo dati*]
- Ad esempio: *ARRAY[0..100] OF LREAL*



Tipi di dati derivati: Array

- L'accesso di una variabile di tipo array si effettua tramite un indice, secondo la seguente sintassi:
 - *[Nome variabile][indice]*
- L'indice può essere una costante o una variabile, ad esempio:
 - *MiaVar[10]*
 - *MiaVar[i]*
- L'indice può anche contenere espressioni matematiche, ma solo in ST:
 - *MiaVar[2*i+k]*

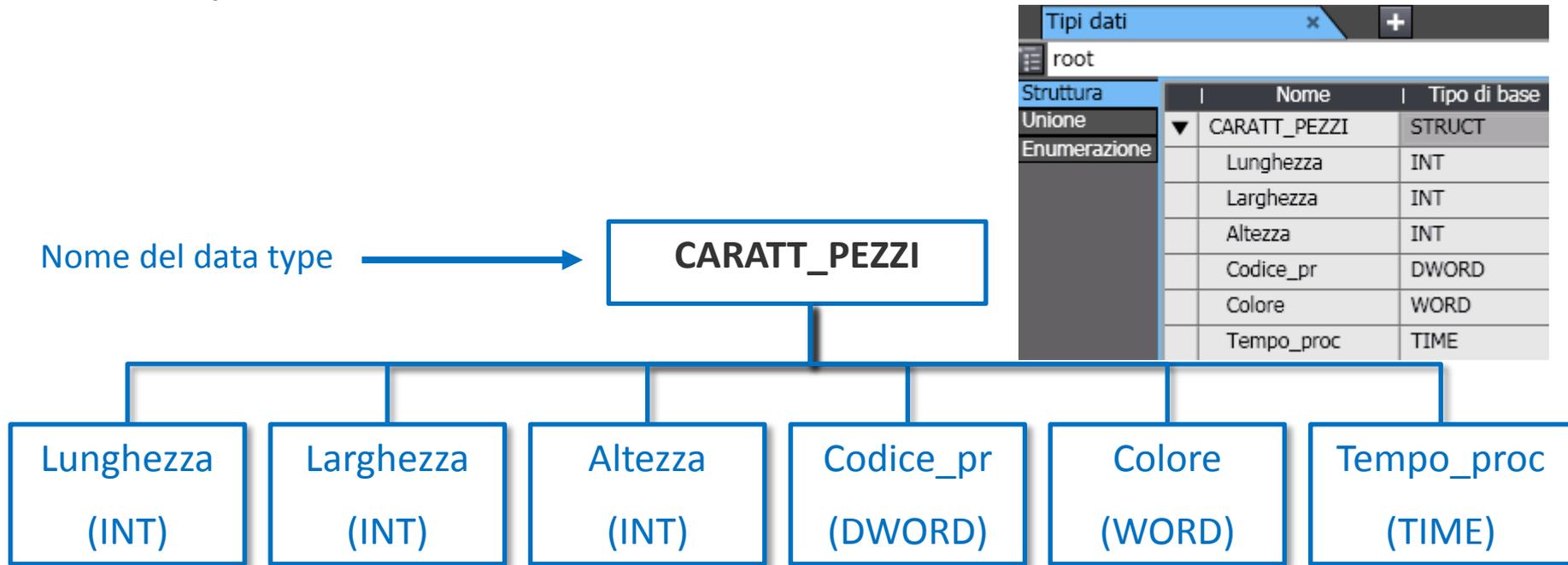
Tipi di dati derivati: Array

Alcune regole:

- E' possibile creare array su una, due o tre dimensioni. Esempio:
ARRAY[0..99,0..49]
- E' possibile creare array di qualsiasi tipo di dati, anche array di istanze di FB (monodimensionali).
- Il primo elemento può corrispondere ad un indice maggiore di 0.
- Un array può avere al massimo 65535 elementi

Tipi derivati: Strutture

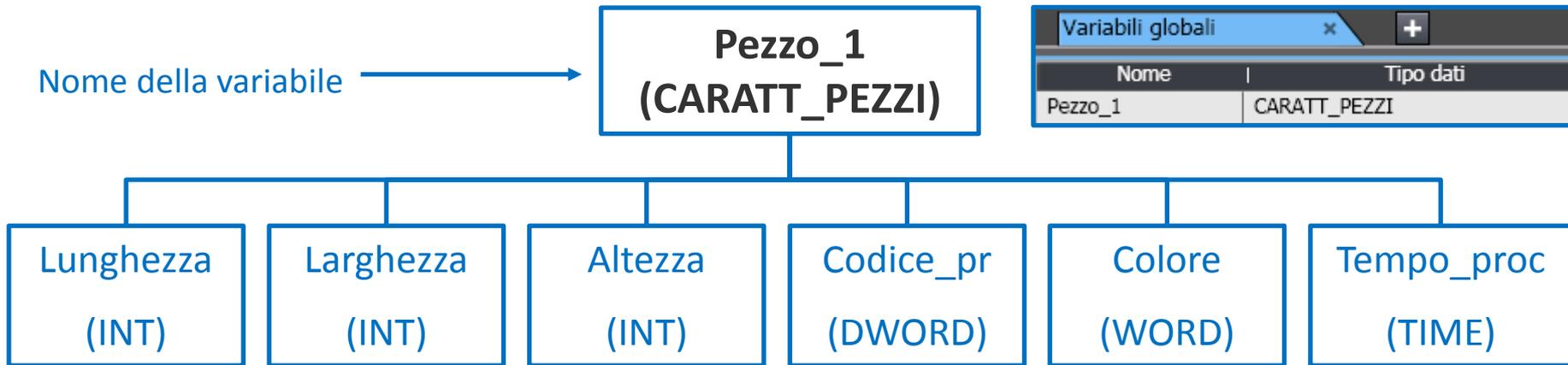
- Servono per strutturare una moltitudine di dati all'interno di un'unica variabile.
- Le strutture possono contenere elementi, ciascuno accessibile attraverso un proprio nome, di tipi di dati diversi tra loro.
- Esempio:



Tipi derivati: Strutture

- Basta dichiarare una sola variabile di tipo “CARATT_PezZI” per avere a disposizione tutte le variabili relative agli elementi della struttura.

Esempio:

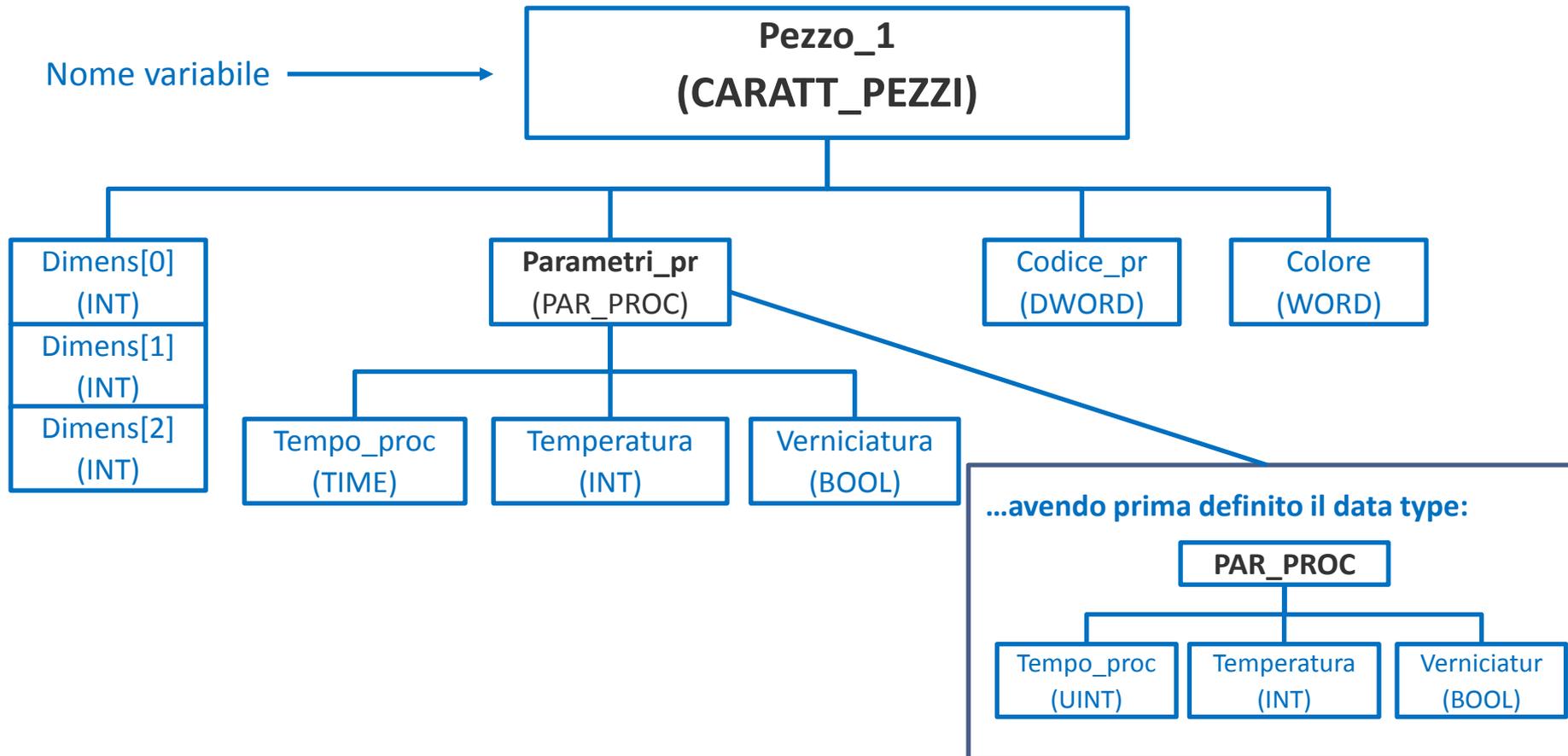


- Per richiamare un elemento della variabile nel programma la sintassi è:

variabile.elemento

Tipi derivati: Strutture

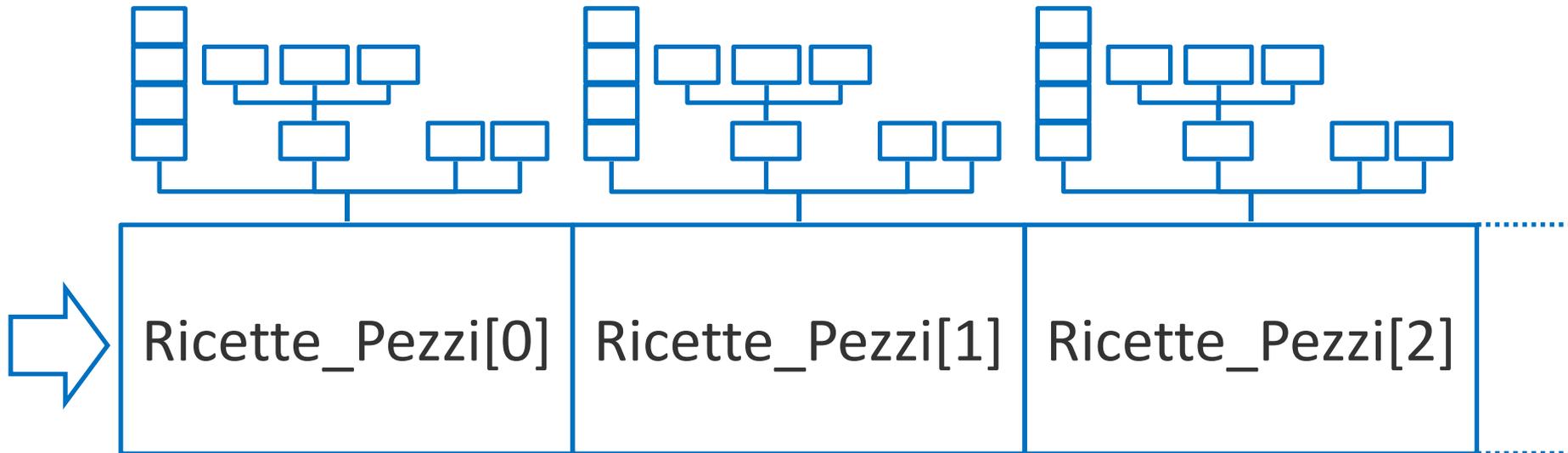
- Gli elementi della struttura possono essere anche tipi derivati come **array**, **sub-range**, **enumerazioni** o **strutture annidate**.
- Esempio:



Tipi derivati: Strutture

- Una volta definito un tipo dati struttura è possibile dichiarare degli **array di strutture**:
- Esempio:

Nome	Tipo dati
Ricette_Pezzi	ARRAY[0..99] OF CARATT_Pezzi



Tipi derivati: Strutture

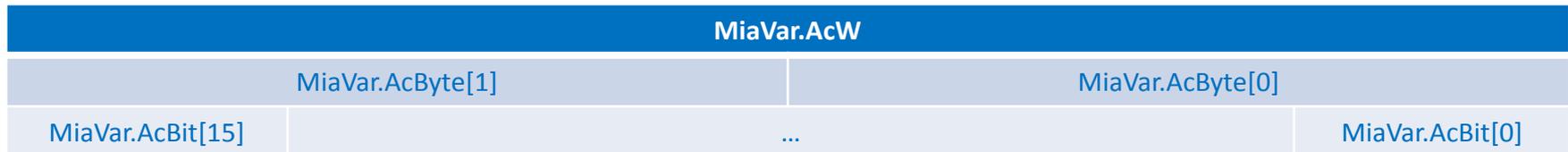
Oggetto	Specifica	
Nome della struttura	Stesse regole relative ai nomi delle variabili	
Tipi di dati degli elementi della struttura	Classificazione	Tipi di dati
	Tipi di dati base	tutti
	Tipi di dati derivati	tutti Nota: L'uso di strutture riflesse o circolari non è consentito
		Array di tipi base o derivati
Proprietà impostabili degli elementi della struttura	Nome dell'elemento, valore iniziale, commento	
Numero di elementi	Tra 1 e 2048	
Livello massimo di annidamento di strutture dentro strutture	Fino ad 8 livelli (I nomi della variabile e degli elementi annidati devono essere tali che il percorso più lungo sia al più di 511 byte)	
Dimensione massima di una variabile struttura	Non limitata	

Tipi derivati: Unioni

- Definiscono l'accesso ad uno stesso dato tramite tipi di dati differenti
- Si applica solo a stringhe di bit (BOOL, BYTE, WORD, DWORD, LWORD e array di questi)
 - Esempio: Accesso ad un dato a 16 bit a word, a byte e a bit

Tipi dati		
root		
Struttura	Nome	Tipo di base
Unione	▼ Dato_a_16_bit	UNION
Enumerazione	AcW	WORD
	AcByte	ARRAY[0..1] OF BYTE
	AcBit	ARRAY[0..15] OF BOOL

Variabili	
Nome	Tipo dati
MiaVar	Dato_a_16_bit



- Nota: le dimensioni di ciascun elemento devono coprire tutti i 16 bit del dato.



DOMANDE?

SYSTMAC
always in control

OMRON

OEE-I



Introduzione Reti

Corso Base Machine Controller

OMRON

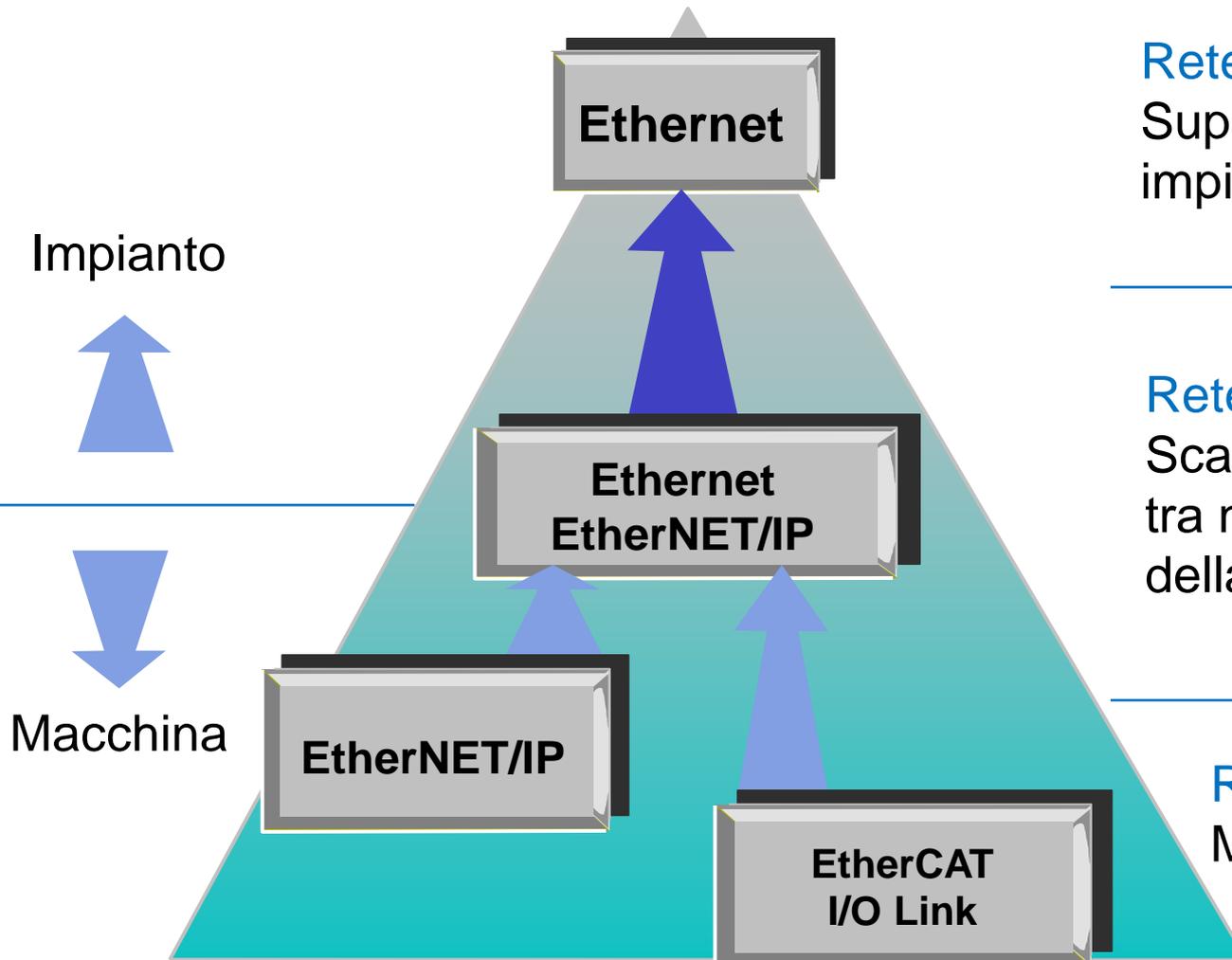
OEE-I

La comunicazione nell'automazione industriale

Le reti di comunicazione nell'ambito dell'automazione industriale sono utilizzate per:

- Integrazione dei processi industriali
- Controllo dei processi
- Pianificazione della produzione
- Manutenzione degli impianti
- Sicurezza

Reti per l'automazione

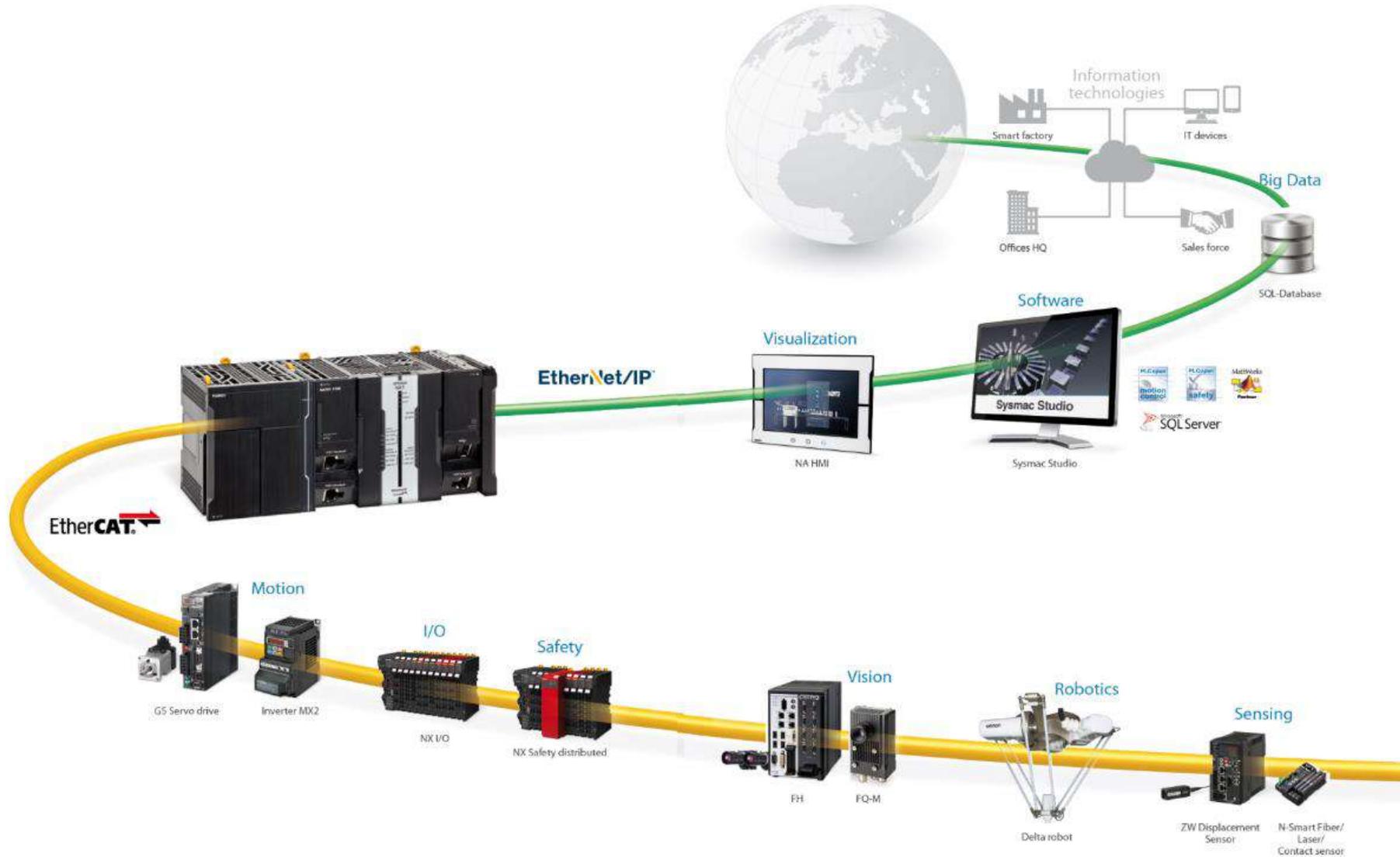


Rete per le informazioni:
Supervisione e gestione impianto

Rete per il Controllo:
Scambio dati tra PLC, tra macchine o all'interno della stessa macchina

Rete per il campo:
Moduli remoti di I/O

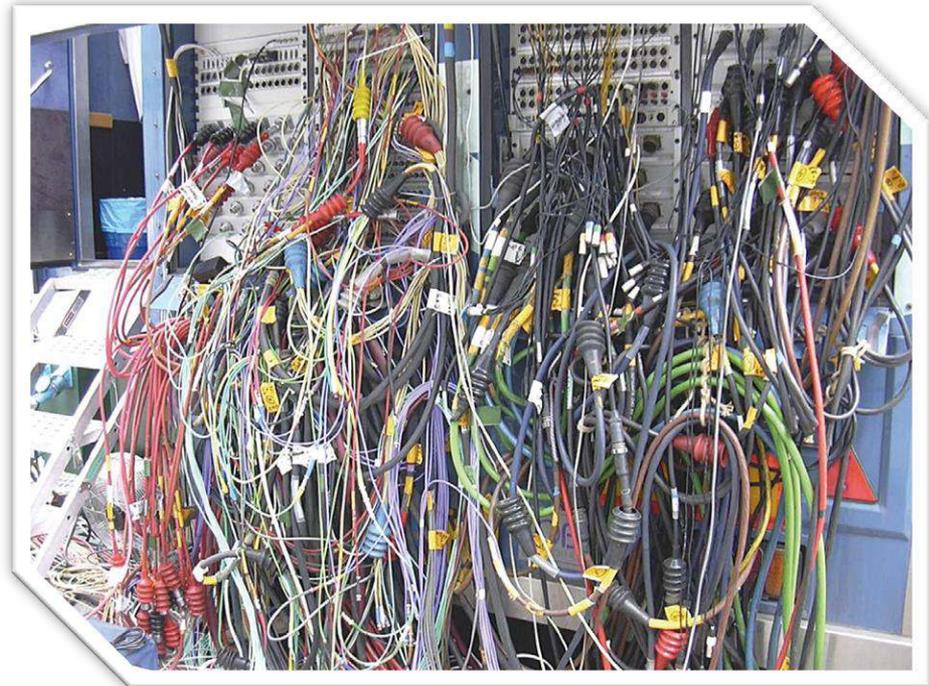
Integrazione verticale



Perche?

Principali benefici:

- Costi di installazione ridotti
- Tempi di messa in servizio inferiori
- Manutenzione semplice e veloce



Rete per le Informazioni

- Comunicazione tra dispositivi dedicati alla Supervisione e alla Gestione dello Stabilimento e dell'Azienda
- I dispositivi utilizzati sono PC e/o Workstation ma è sempre più diffuso l'uso di PLC.
- Le informazioni trasmesse sono complesse, e composte da molti byte, spesso si tratta di interi file di dati
- Le frequenze di trasmissione non sono alte e non è necessario assicurare la trasmissione entro intervalli di tempo critici
- Esempio: OPC/UA, Ethernet/IP o TCP/IP

Rete per il Controllo

- Comunicazione tra dispositivi dedicati al controllo degli impianti e alla supervisione di macchina.
- I dispositivi utilizzati sono PC, PLC e HMI.
- Le informazioni trasmesse non sono complesse.
- Le frequenze di trasmissione sono alte.
- E' necessario assicurare la trasmissione entro intervalli di tempo critici (in pratica, in tempo reale).
- Esempio: Ethernet/IP

Rete per il Campo (fieldbus)

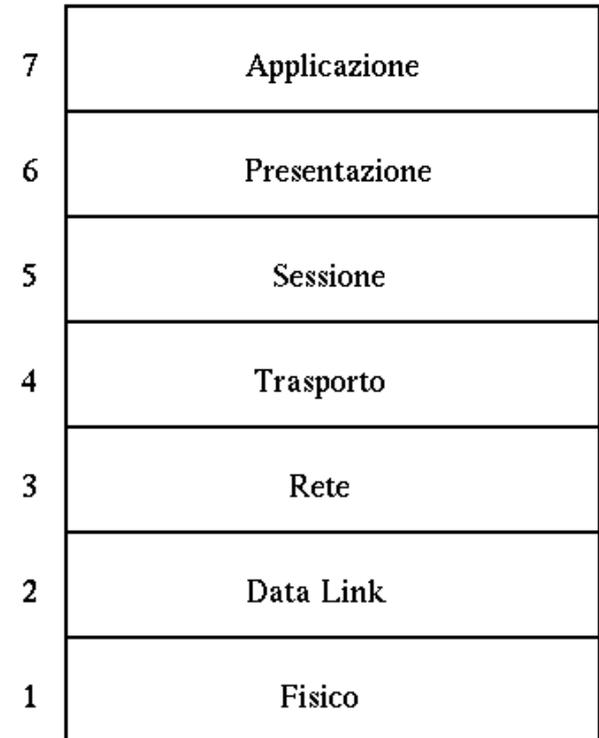
- Comunicazione tra dispositivi dedicati al controllo degli impianti, e all'area dei sensori/attuatori
- I dispositivi utilizzati sono PLC, sensori, sensori “intelligenti”, attuatori ed attuatori “intelligenti”, morsettiere di I/O.
- Le informazioni trasmesse sono molto semplici, dell'ordine dei bit o di pochi byte.
- Le frequenze di trasmissione sono molto alte, nell'ordine dei millisecondi.
- Esempio: EtherCAT

Il protocollo di comunicazione

- Il protocollo di comunicazione è l'insieme di regole e comportamenti che due entità distinte devono rispettare per scambiare informazioni tra loro.
- La comunicazione può avvenire a differenti livelli (fase intermedie di gestione del messaggio).
- Lo standard ISO 7498 definisce il modello OSI (open system interconnection) in cui viene fissata una struttura a 7 livelli (layer) come modello di riferimento per lo sviluppo di protocolli orientati all'interconnessione di sistemi aperti.

Il modello ISO/OSI

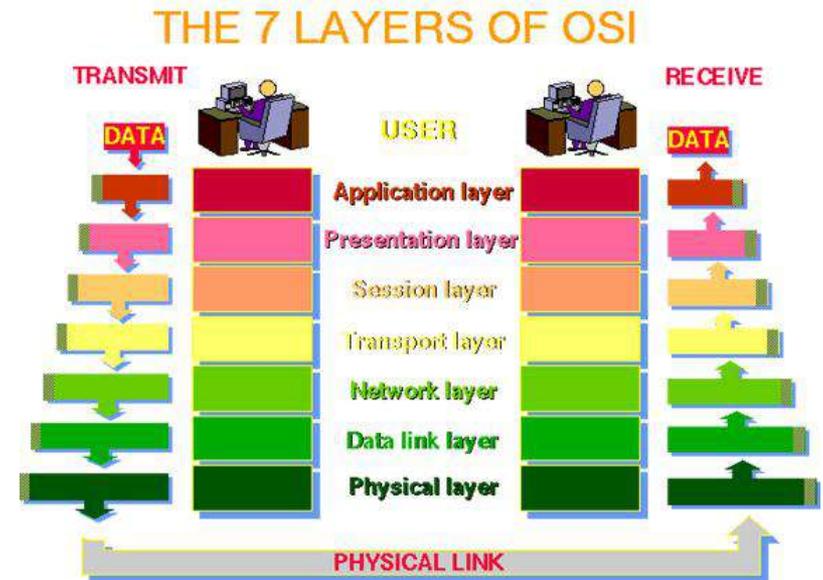
- Questo modello definisce 7 livelli (logici). Il più basso è il livello di comunicazione fisica, il più alto è il programma applicativo.
- Non tutti sono effettivamente necessari, ma comunque servono da riferimento.



I Livelli ISO/OSI

Il modello ISO/OSI

- In ogni livello vengono aggiunte informazione per la gestione del processo di comunicazione (PDU: protocol data unit). Questa procedura si chiama procedura di incapsulamento
- Tra due livelli esiste una relazione di tipo Client/Server l'interfaccia tra di essi si chiama SAP (Service Access Point)
- Il livello fisico è l'unico livello che realizza una comunicazione "orizzontale"

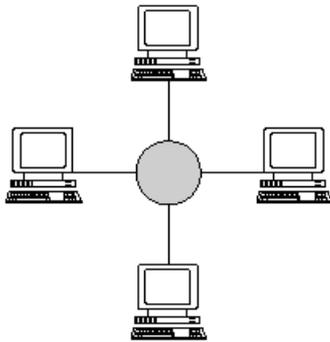


Modello OSI e reti industriali

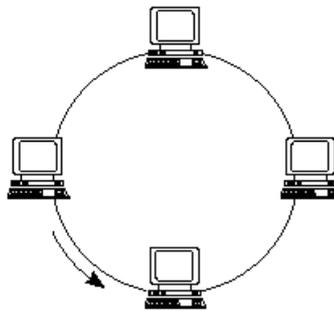
- Il modello OSI è stato pensato per interconnessioni fra sistemi informatici.
- Ad ogni livello viene eseguita una operazione sui dati quindi è chiaro che il modello comporta un peso sensibile in termini di performance e risorse impegnate (al data link vengono aggiunti i PDU di ogni livello riducendo il throughput)
- I sistemi orientati a elevate performance (bassi tempi di latenza, determinismo ed elevato throughput) fanno riferimento all'architettura EPA (Enhanced Performance Architecture) in cui si mantengono solo 3 livelli: fisico, data link e applicazione (1, 2 e 7).

Topologia

- Disposizione dei collegamenti di una rete di comunicazione.
- Topologie tipiche: Bus, Stella, Anello, Mista.



Stella



Anello



Bus



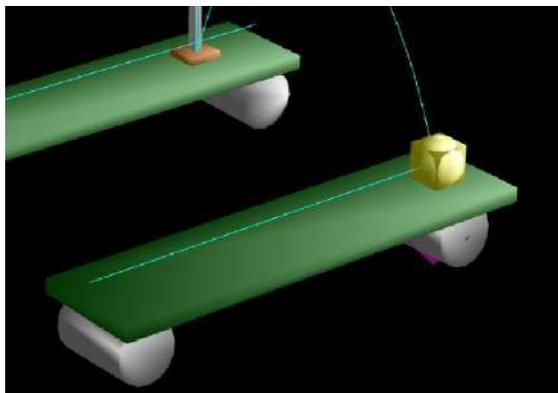
DOMANDE?



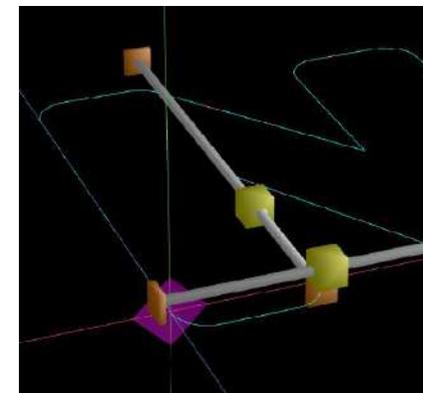
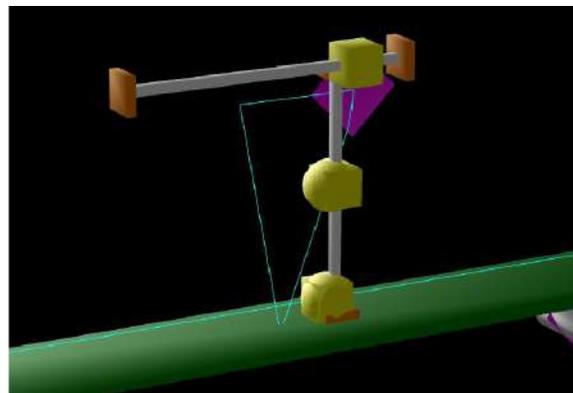
Introduzione al modulo MC

Il modulo Motion Control

- Il modulo software Motion Control, integrato in ogni CPU NJ, ha il compito di gestire gli Assi della macchina:
- Controlla fino ad un totale di 64 assi fisici e virtuali
- È in grado di effettuare movimentazioni
 - **Discrete:** movimenti singoli assi
 - **Sincronizzate:** Cam o Gear
 - **Coordinate:** mediante la creazione di Gruppi di Assi



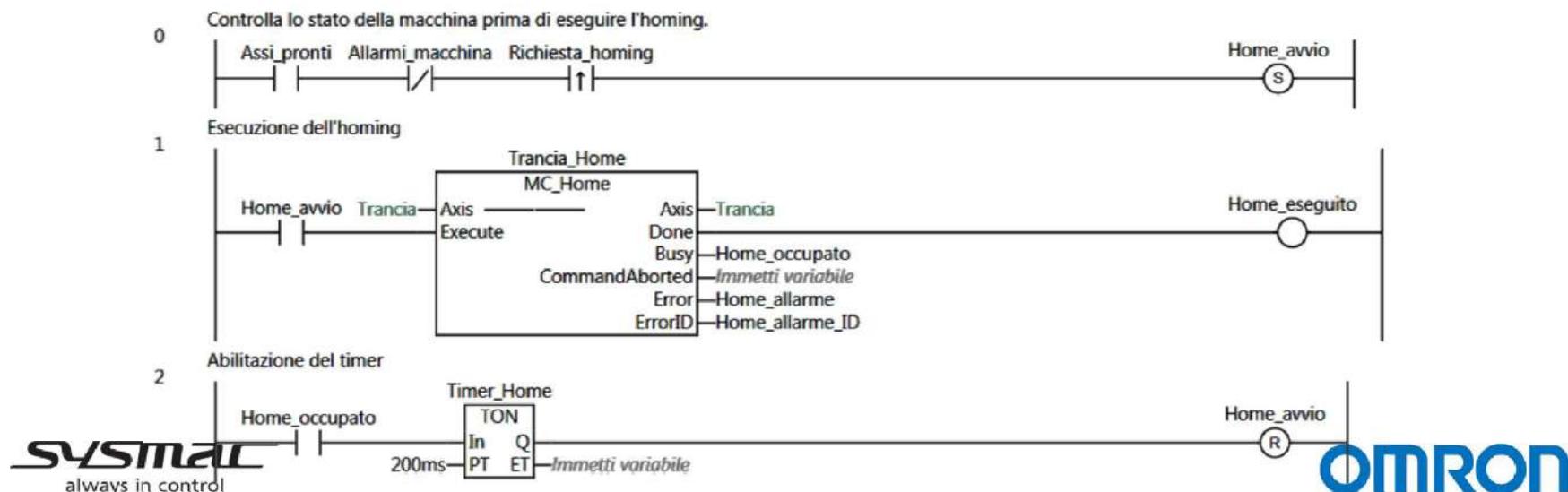
SYSTMAC
always in control



OMRON

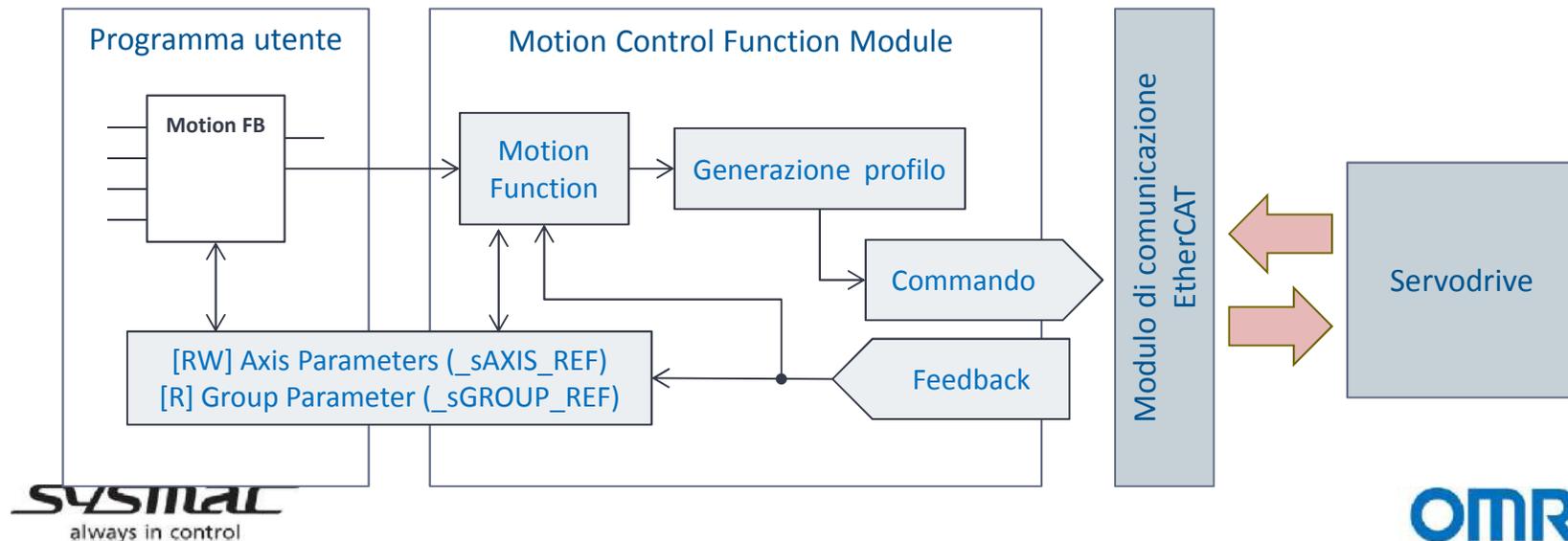
Il modulo Motion Control

- La libreria Motion FB, certificata IEC 61131-3, è composta da FB certificati PLCOpen e da FB tipici Omron
- Le istruzioni Motion sono scritte nei linguaggi Ladder ed Structured Text (ST) e sono inserite all'interno del programma IEC, il quale integrerà sia le istruzioni Motion e sia la logica generica del programma stesso



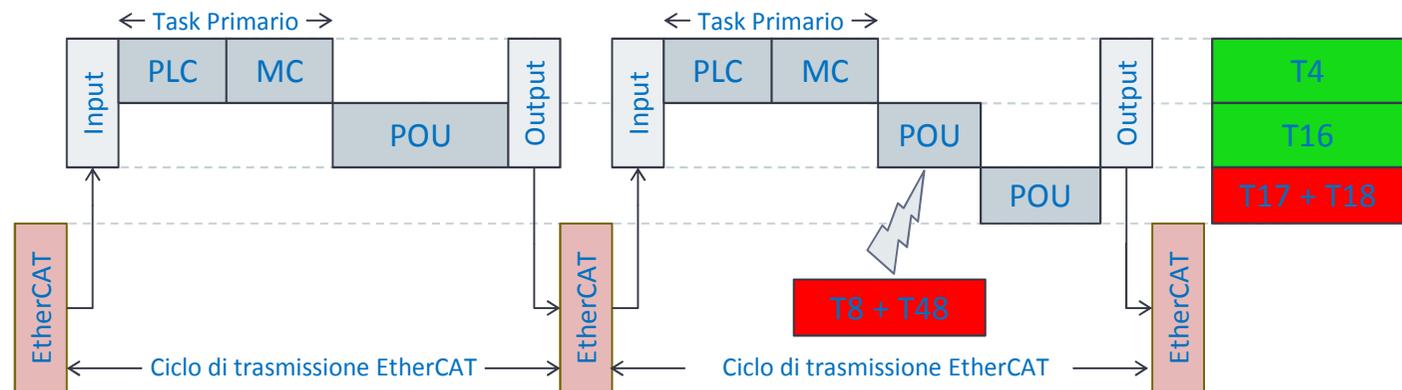
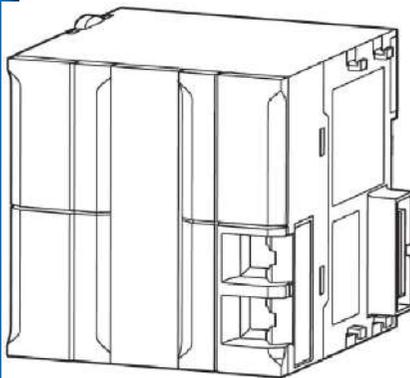
Motion Control Function Module ed EtherCAT

- Il Motion Control Function Module si occupa di controllare gli assi e generare i profili di movimento
- Il motion Control Function Module è inserito nel task IEC primario, sincrono alla comunicazione EtherCAT.
- I parametri asse sono collegati ad una struttura dati di tipo `_sAXIS_REF`, che permette il monitoraggio nel programma utente
- I parametri dei gruppi asse sono collegati ad una struttura dati di tipo `_sGROUP_REF`, che permette il monitoraggio nel programma utente



Motion Control Function Module ed EtherCAT

- I comandi al Motion Control Function Module possono essere dati nei task con priorità 4 e 16
- I comandi al Motion Control Function Module NON possono essere dati nei task con priorità 17 e 18
- I comandi al Motion Control Function Module NON possono essere dati negli event task con priorità 8 e 48





Configurazione dell'asse

SYSTMAC
always in control

OMRON

L'asse EtherCAT

- NJ è stato progettato per realizzare complesse operazioni motion tra azionamenti EtherCAT G5. Il modulo MC è in grado di generare profili, controllando posizione, velocità, coppia e jerk di tutti gli assi collegati.
- Il controllore ed il programma IEC sono sincronizzati per mezzo di variabili assi, in sola lettura, di tipo `_sAXIS_REF`.

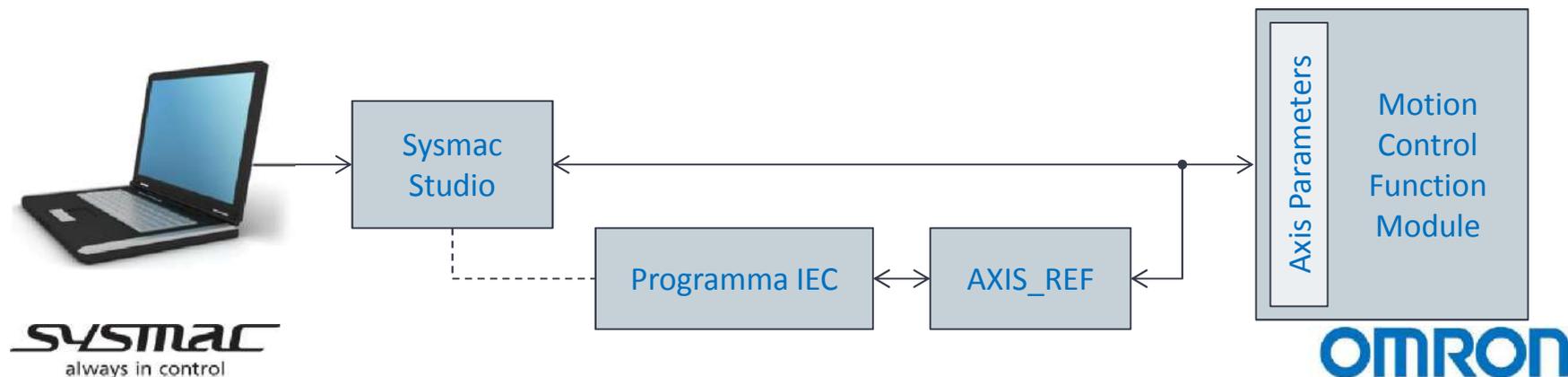


SYSTMAC
always in control



Variabili asse e tipo di dato `_sAXIS_REF`

- Le Variabili asse, di tipo `_sAXIS_REF`, contengono:
 - Parametrizzazione asse
 - Comandi da inviare al drive
 - Feedback (stato dell'asse, posizione, velocità, errore, etc)
- Sysmac Studio accede in modo diretto alla variabile, durante la fase di compilazione, per generare la configurazione
- Durante il runtime, il programma IEC fa riferimento, alla variabile asse, di tipo, `_sAXIS_REF` per monitorare gli assi



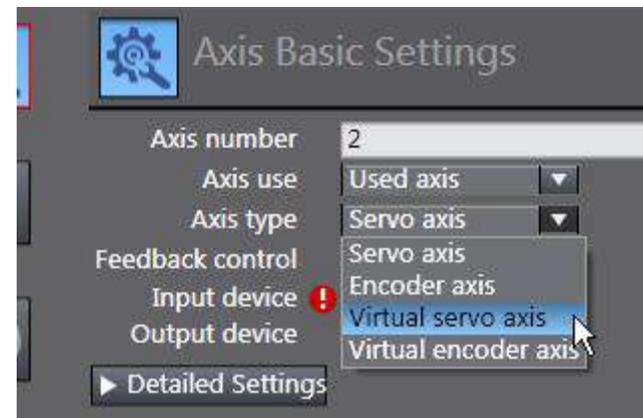
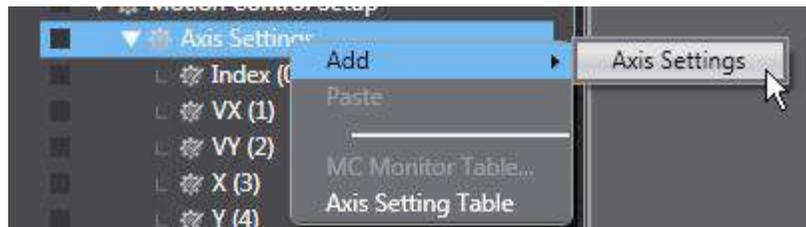
Tipi di assi

- NJ501 supporta i seguenti tipi di assi:

Tipo di asse	Descrizione
Servoazionamento	Asse utilizzato per controllare slave Servo Drive in EtherCAT (comando + feedback).
Encoder	Asse utilizzato per controllare ingressi encoder in EtherCAT (solo feedback).
Servoazionamento virtuale	Asse servo simulato, presente solamente nel Modulo MC, utilizzato, ad es., come master nelle sincronizzazioni (comando + feedback simulato).
Encoder virtuale	Asse encoder simulato, utilizzato solamente per debugging (solo feedback simulato). Simula encoder esterno

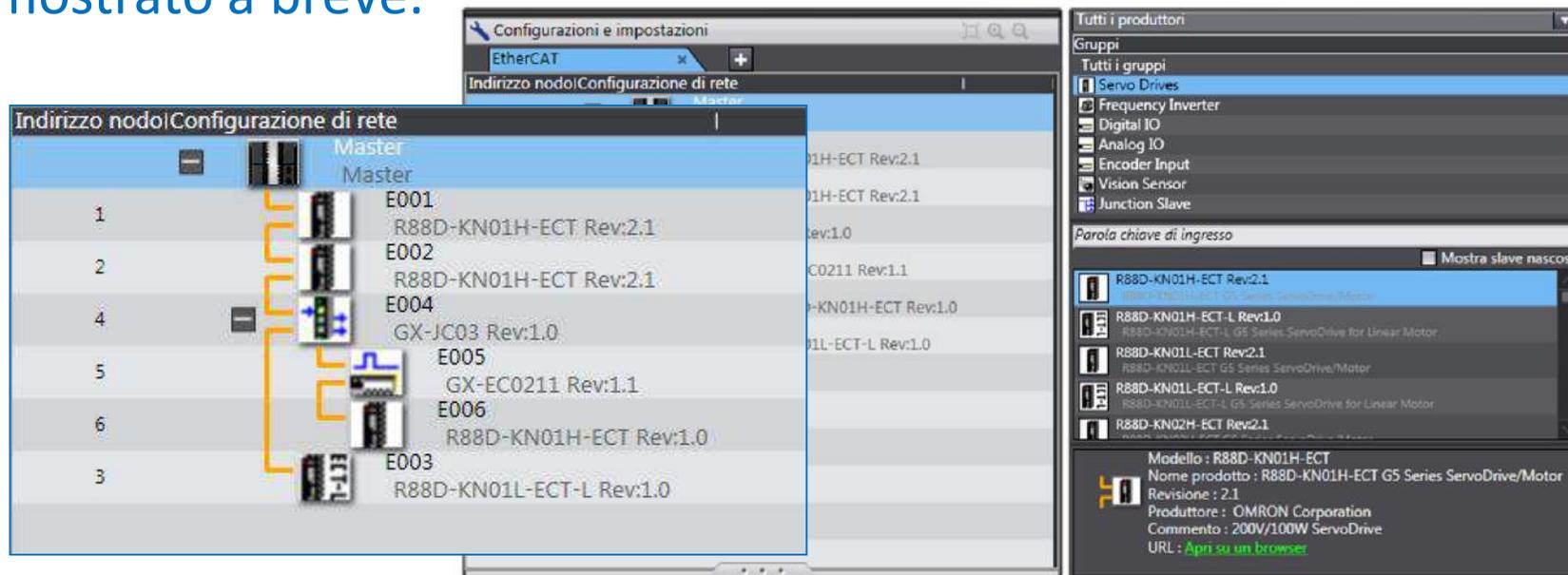
Configurazione assi Logici - Sysmac Studio

- Gli assi logici della macchina sono utilizzati dal motion function module e dal programma utente.
- Gli assi logici possono essere associati ad ogniuna delle quattro tipologie di asse presentate nella precedente slide tramite le 'Impostazioni base asse' (Axis Basic Setting), come mostrato a breve.



Configurazione assi fisici - Rete EtherCAT

- Per configurare la struttura della rete EtherCAT occorre inserire tutti gli assi fisici utilizzati dalla macchina.
- Poi si associa ad ogni asse fisico il corrispondente asse logico, tramite le 'Impostazioni base asse' (Axis Basic Setting), come mostrato a breve.



Impostazioni dell'asse

Tasto	Nome	Descrizione
	Impostazioni base asse	Settaggi relativi al tipo di asse e al nodo da assegnare
	Impostazioni di conversione del modulo	Settaggi relativi al rapporto di riduzione elettronica
	Impostazioni di funzionamento	Settaggi relativi ai limiti massimi di velocità, accelerazione e coppia e impostazioni di monitoraggio
	Altre impostazioni di funzionamento	Settaggi relativi al comportamento degli I/O del servodrive
	Impostazioni limite	Settaggi relativi ai limiti software e al following error
	Impostazioni approccio diretto	Settaggi relativi alla procedura di homing dell'asse
	Impostazioni contatore posizione	Settaggi relativi al metodo di acquisizione della posizione (asse lineare o rotativo)
	Impostazioni servoazionamento	Settaggi relativi al metodo di calcolo della posizione interna al servoazionamento

Impostazioni base asse

• Questa pagina è utilizzata per effettuare le impostazioni iniziali di configurazione dell'asse:



- Numero dell'asse (per l'allocazione nella struttura `_MC_AX`)
- Tipo di asse
- Utilizzo dell'asse (utile per rendere lo stesso programma compatibile a differenti configurazioni degli assi)
- Controllo Feedback: (futuro)
- Nodo EtherCAT dell'eventuale dispositivo fisico connesso all'asse

Axis number	0		
Axis use	Used axis		
Axis type	Virtual servo axis		
Feedback control	No control loop		
Input device 1	<Not assigned>	Channel	
Input device 2	<Not assigned>	Channel	
Input device 3	<Not assigned>	Channel	
Output device 1	<Not assigned>	Channel	
Output device 2	<Not assigned>	Channel	
Output device 3	<Not assigned>	Channel	
▶ Detailed Settings			

Impostazioni di conversione del modulo

• Questa pagina è utilizzata per impostare il rapporto di riduzione elettronica tra il controllore ed il servosistema:



- Modulo di visualizzazione, per una più chiara comprensione delle movimentazioni in fase di programmazione e di test
- Impulsi di comando per rotazione del motore
- Distanza del carico prodotta da una rotazione del motore

Modulo di visualizzazione impulso mm um nm gradi pollice

Contatore di impulsi di comando per rotazione del motore impulso/rev

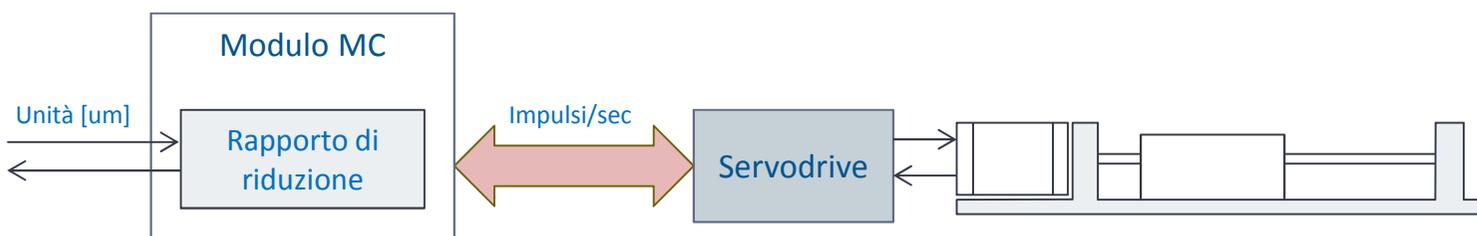
Distanza corsa di lavoro per rotazione del motore um/rev

Condizioni: Formula conversione modulo

Numero di impulsi [impulso] = $\frac{\text{Contatore di impulsi di comando per rotazione del motore (UDINT)}}{\text{Distanza corsa di lavoro per rotazione del motore (LREAL)}} * \text{Distanza corsa [Modulo di visualizzazione]}$

Impostazioni di conversione del modulo

- La conversione di unità di misura verrà effettuata a bordo del modulo MC, in modo da fornire il comando scalato in maniera corretta al servodrive.



Esempio.

Modulo di visualizzazione: [mm]

Impulsi encoder/giro = $2^{17} = 131072$

Impulsi di comando = $131072 * 5$

Distanza corsa di lavoro = $10 * 2$

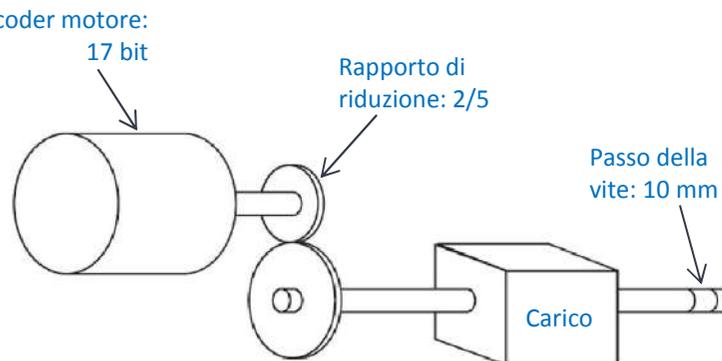
Questo modo di calcolare permette di evitare valori irrazionali o periodici, non rappresentabili o arrotondati nell'impostazione dell'asse

In questo caso è possibile usare i valori riportati di seguito

Impulsi di comando = 131072

Distanza corsa di lavoro = $10/5 * 2 = 4$

in quanto 'Distanza corsa di lavoro' è un numero rappresentabile senza approssimazione



Impostazioni di funzionamento

- In questa pagina è possibile inserire le impostazioni relative a:
 - Valori massimi di velocità, accelerazione e decelerazione
 - Valori di coppia, velocità, accelerazione e decelerazione per i quali viene segnalato il superamento
 - Intervallo di precisione del posizionamento (in movimentazione standard e durante l'homing)



▼ Velocità/Accelerazione/Decelerazione				
Velocità massima	415	mm/s	Valore di avviso velocità	50 %
Velocità jog massima	50	mm/s		
Accelerazione massima	844000	mm/s ²	Valore di avviso accelerazione	25 %
Decelerazione massima	422000	mm/s ²	Valore di avviso decelerazione	25 %
Sovraccelerazione/sovradecelerazione	Arresto errore non grave ▼			
Selezione del funzionamento all'indietro	Arresto immediato ▼			
▼ Coppia				
Valore di avviso coppia positiva	250	%	Valore di avviso coppia negativa	120 %
▼ Monitoraggio				
Intervallo in posizione	0.1	mm	Tempo di controllo in posizione	4 ms
Costante di tempo filtro velocità effettiva	0	ms	Intervallo posizione zero	10 mm

Sovraccelerazione/sovradecelerazione

- Questa opzione agisce quando lo spazio necessario per effettuare il blending è maggiore dello spazio fra la posizione attuale (al momento della messa in coda dell'istruzione) e la destinazione

1. Blending changed to buffered

- Il movimento non viene raccordato. L'asse si ferma alla destinazione e riparte

2. Use rapid acceleration/deceleration

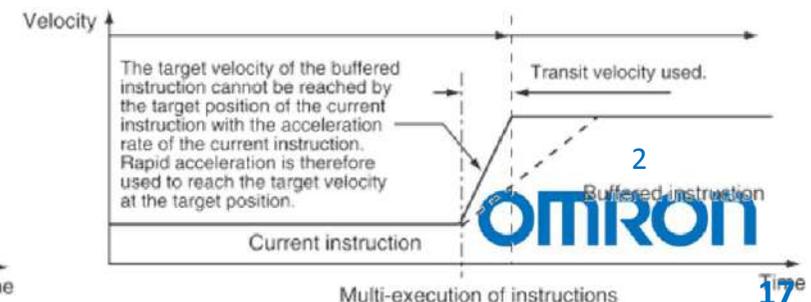
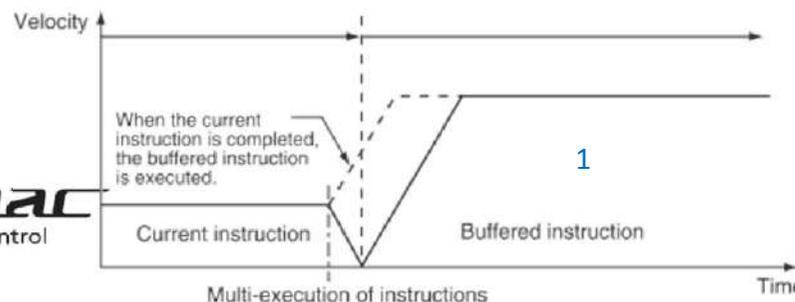
- Il movimento viene raccordato. E' calcolata l'acc/dec necessaria per arrivare a destinazione dalla posizione attuale. Questo valore è usato nella generazione del profilo di movimento

3. Minor Fault stop

- Viene generato un Minor Fault e l'asse si ferma. La posizione finale potrebbe non essere la destinazione richiesta



SYSTMAL
always in control



Selezione del funzionamento all'indietro

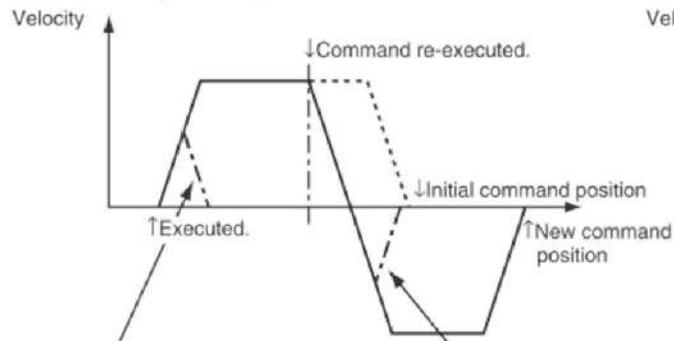
- Questa opzione agisce quando si ha una inversione nella direzione di moto.
- Deceleration Stop
- L'inversione di moto viene effettuata usando le acc/dec impostate
- Immediate Stop
- L'inversione di moto viene effettuata nel momento in cui il comando è ricevuto, senza usare acc/dec



SYSTRAC
always in control

● When a Reverse Turn Occurs for the New Command Value

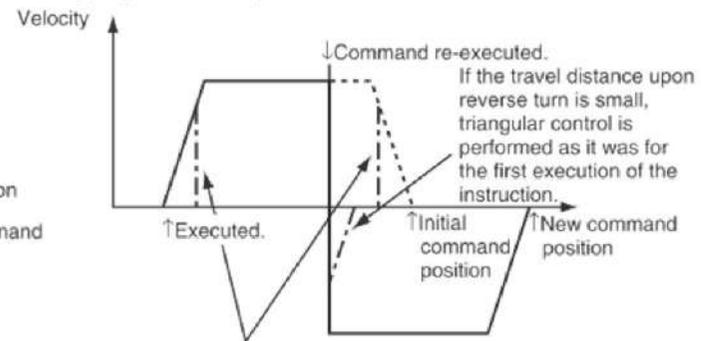
Decelerating to a Stop after Reverse Turn



If the instruction is re-executed during acceleration, the axis starts deceleration as soon as the instruction is re-executed.

If the travel distance upon reversal is small, triangular control is performed as it was for the first execution of the instruction.

Stopping Immediately after Reverse Turn

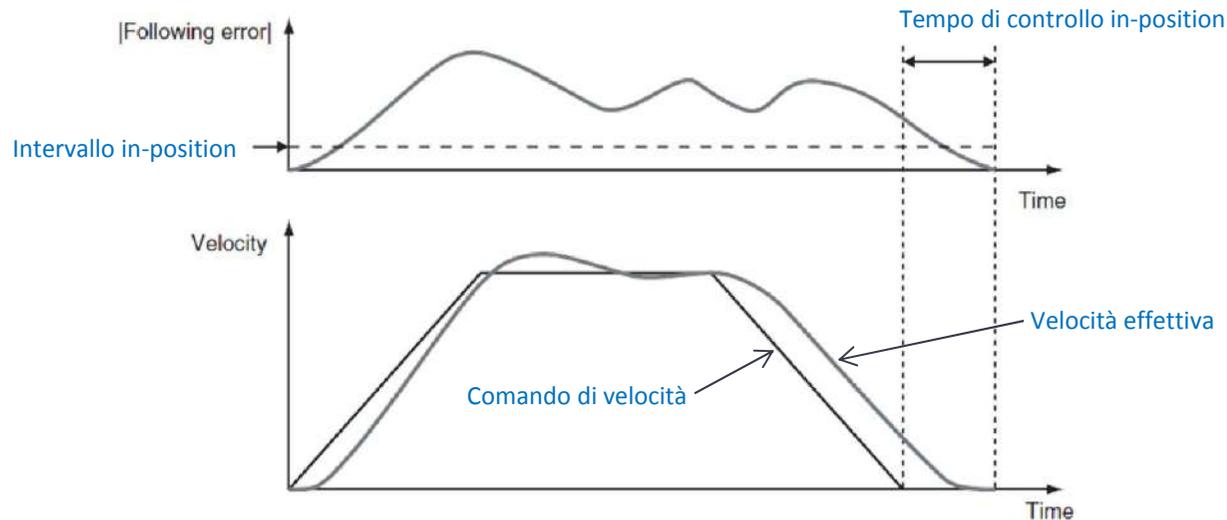


If the instruction is re-executed during acceleration or deceleration, the axis stops immediately upon re-execution. This also occurs during deceleration.

OMRON

Impostazioni di funzionamento

- Quando la differenza tra posizione di target e quella effettiva (errore d'inseguimento) è inferiore all'intervallo in-position, il posizionamento è considerato ultimato.



Se l'errore d'inseguimento rimane superiore all'intervallo in-position per un tempo superiore al tempo di controllo in-position, viene generato un allarme.



Altre impostazioni di funzionamento

- In questa pagina è possibile definire i metodi di arresto sull'ingresso di arresto immediato E-STP e sugli ingressi di finecorsa (P-OT e N-OT) del servodrive.
- È possibile, inoltre, definire eventuali limiti di coppia attivi durante le operazioni di movimentazione.
- E' possibile configurare NC o NO i sensori di emergenza, extracorsa e Home



Immediate stop input stop method	Immediate stop	▼
Limit input stop method	Immediate stop	▼
Drive error reset monitoring time	200	ms
Maximum positive torque limit	300.0	%
Maximum negative torque limit	300.0	%
Immediate stop input logic inversion	Do not invert	▼
Positive limit input logic inversion	Do not invert	▼
Negative limit input logic inversion	Do not invert	▼
Home proximity input logic inversion	Do not invert	▼

Impostazioni limite

- In questa pagina è possibile abilitare i limiti software alla corsa dell'asse: questi vengono di norma utilizzati in accoppiata con i finecorsa per prevenire potenziali danneggiamenti al macchinario, dovuti ad imperizie nella programmazione o nell'uso della macchina.
- È possibile, inoltre, impostare i valori limite dell'errore di inseguimento, che generano un messaggio di errore.

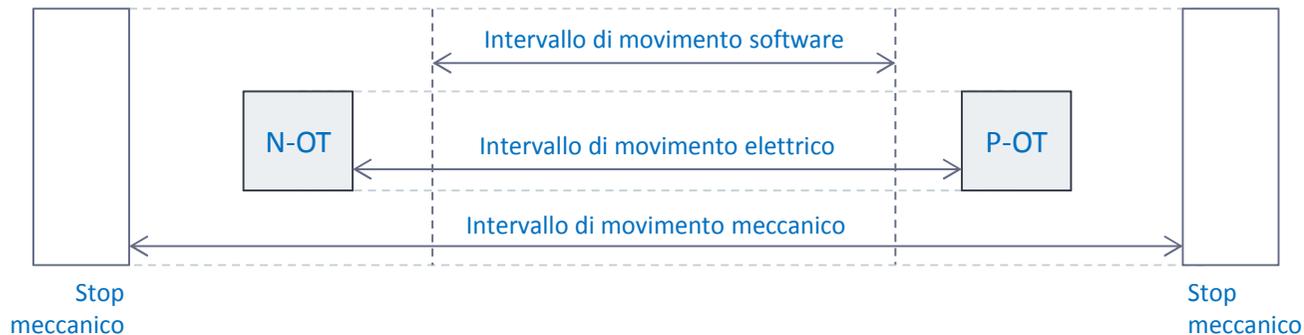


▼ Limite software		
Limiti software	Arresto decelerazione per posizione effettiva ▼	
Limite software positivo	2000	mm
Limite software negativo	500	mm
▼ Deviazione posizione		
Valore eccessivo deviazione posizione	6	mm
Valore di avviso deviazione posizione	3	mm

Impostazioni limite

- Durante il normale funzionamento della macchina, se vengono attivati i limiti software, il movimento dell'asse è permesso in due soli casi:

- L'asse si trova entro i limiti software impostati
- L'asse si trova al di fuori dei limiti software, ma si muove in direzione del limite stesso



- I limiti software possono essere abilitati quando la modalità di conteggio della posizione è lineare.



Impostazioni approccio diretto

- In questa pagina è possibile selezionare la procedura di homing dell'asse.



▼ Metodo approccio diretto

Metodo approccio diretto: Inversione indietro prossimità/ingresso prossimità approccio diretto ON

Segnale di ingresso approccio diretto: Usa ingresso fase Z come approccio diretto

Direzione avvio approccio diretto: Direzione positiva

Direzione rilevamento ingresso approccio diretto: Direzione negativa

Selezione funzionamento durante ingresso limite positivo: Inversione indietro/arresto immediato

Selezione funzionamento durante ingresso limite negativo: Inversione indietro/arresto decelerazione

Segnale prossimità approccio diretto
Ingresso fase Z

Ingresso limite positivo

Ingresso limite negativo

Avvia da lato negativo del segnale di prossimità dell'approccio diretto: Fine normale

Avvia da attivazione segnale di prossimità dell'approccio diretto: Fine normale

Avvia da lato positivo del segnale di prossimità dell'approccio diretto: Fine normale

▼ Velocità/Accelerazione/Decelerazione

Velocità approccio diretto	25 mm/s	Velocità approccio diretto	10 mm/s
Accelerazione approccio diretto	0 mm/s ²	Decelerazione approccio diretto	0 mm/s ²
Raccordo approccio diretto	0 mm/s ³		

▼ Varie

Distanza maschera ingresso approccio diretto	10000 mm	Offset approccio diretto	-30 mm
Tempo di mantenimento approccio diretto	100 ms		
Valore compensazione approccio diretto	0 mm	Velocità compensazione approccio diretto	10 mm/s

Impostazioni approccio diretto

• Nella parte bassa della pagina, è possibile definire alcuni parametri caratteristiche dell'homing:

- Velocità, accelerazioni e jerk di homing e di prossimità
- Eventuale offset dal punto di fermata del motore e lo zero dell'asse
- Parametri per un eventuale compensazione dello zero
- Tempo di holding e distanza di maschera, utilizzate in alcuni metodi di homing specifici



▼ Velocità/Accelerazione/Decelerazione			
Velocità approccio diretto	25 mm/s	Velocità approccio diretto	10 mm/s
Accelerazione approccio diretto	0 mm/s ²	Decelerazione approccio diretto	0 mm/s ²
Raccordo approccio diretto	0 mm/s ³		
▼ Vane			
Distanza maschera ingresso approccio diretto	10000 mm	Offset approccio diretto	-30 mm
Tempo di mantenimento approccio diretto	100 ms		
Valore compensazione approccio diretto	0 mm	Velocità compensazione approccio diretto	10 mm/s

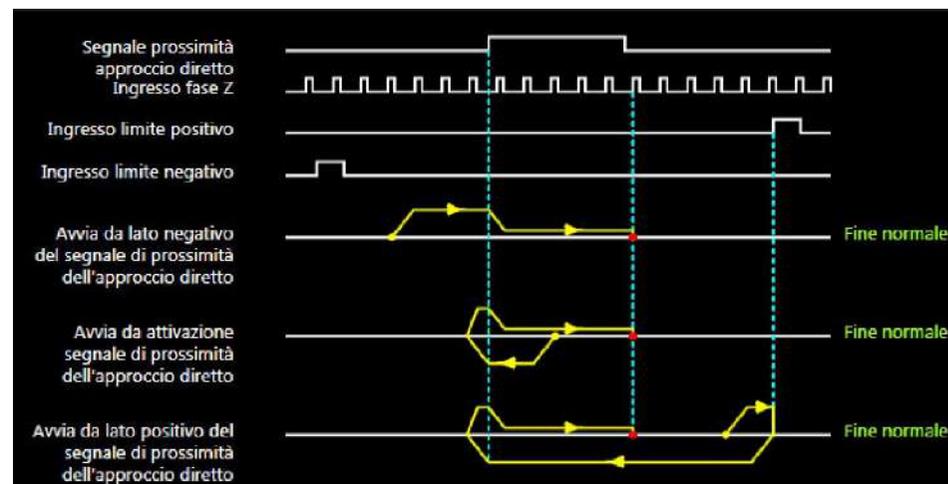
Impostazioni approccio diretto

È possibile impostare metodi di homing con:

- Segnale di prossimità per ridurre la velocità e garantire precisione maggiore
- Segnali finecorsa che invertono la direzione di marcia, in modo da rimanere all'interno dell'area di lavoro
- Ingresso di homing (esterno o fase Z) che ferma il motore



SYSTMAC
always in control

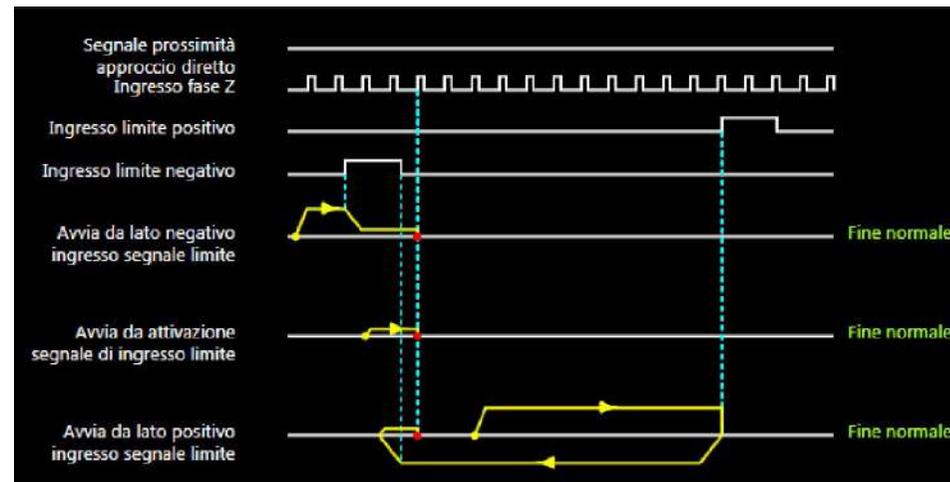


OMRON

Impostazioni approccio diretto

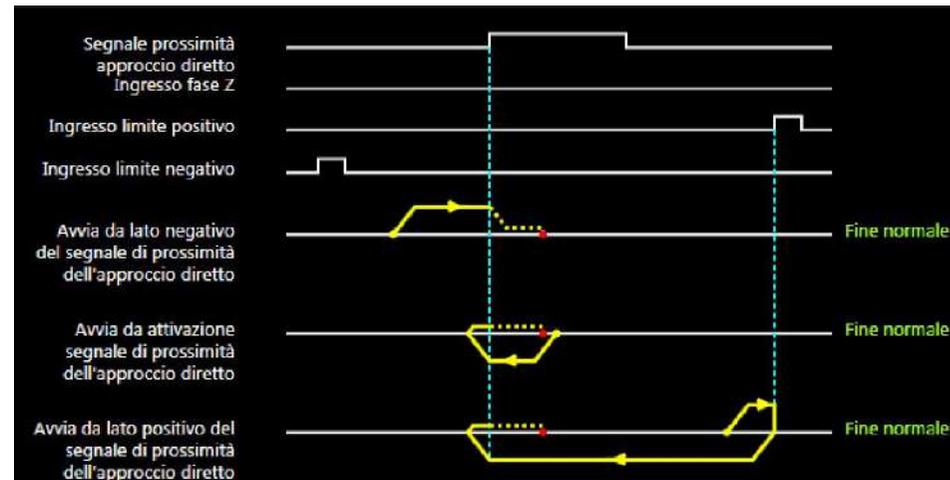
•È possibile impostare metodi di homing con:

- Segnali finecorsa che invertono la direzione di marcia e riducono la velocità, agendo come segnale di prossimità
- Ingresso di homing (esterno o fase Z) che ferma il motore



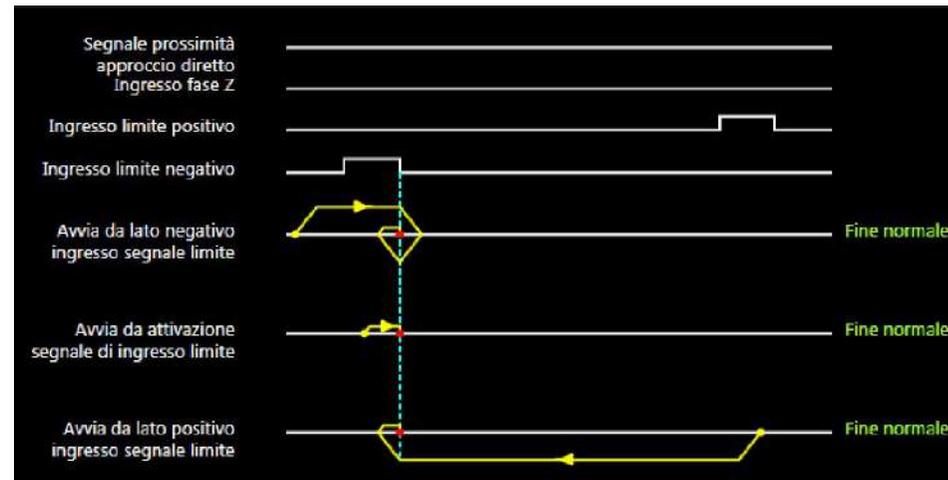
Impostazioni approccio diretto

- È possibile impostare metodi di homing con:
 - Segnale di prossimità che riduce la velocità
 - Fermata del motore che avviene in modo autonomo, senza segnale homing, quando viene trascorso un tempo di Holding dopo il fronte del segnale di prossimità



Impostazioni approccio diretto

- È possibile impostare metodi di homing mediante l'uso dei soli sensori di finecorsa, che agiscono sia da segnali di prossimità (riducendo la velocità) sia da segnali di homing (fermando il motore).



Impostazioni contatore posizione

• In questa pagina è possibile scegliere il metodo di conteggio della posizione:



- Modalità lineare, il limite della corsa sarà dato dai finecorsa e dai limiti software

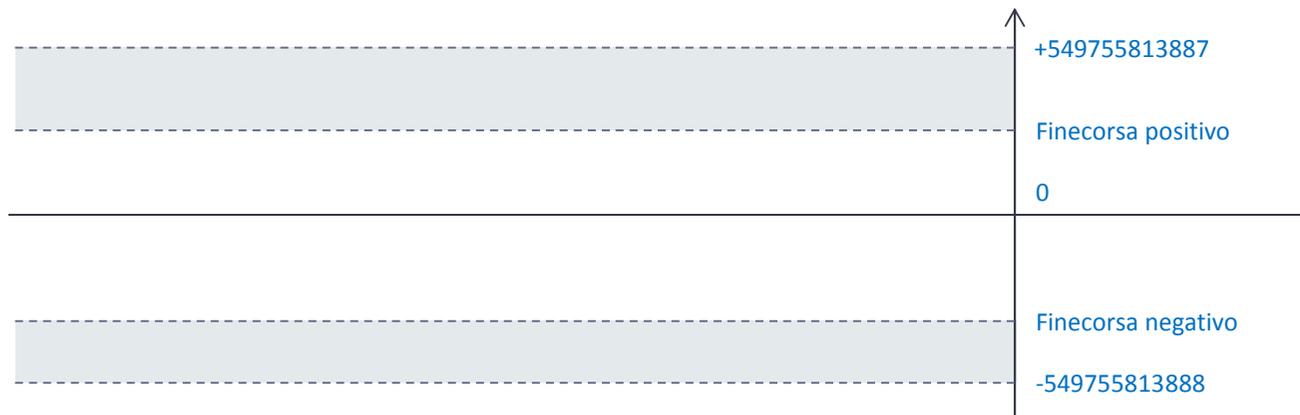
- Modalità rotativa, la corsa non ha limite, quindi il valore che assume la posizione dovrà essere ristretto ad un range ammissibile

- È possibile anche andare a selezionare il tipo di encoder montato a bordo del servomotore.

Modalità conteggio	<input type="radio"/>	Modalità lineare	<input checked="" type="radio"/>	Modalità rotativa
Valore impostazione posizione massimo modulo		180	gradi	
Valore impostazione posizione minima modulo		-180	gradi	
Tipo encoder		Encoder incrementale		

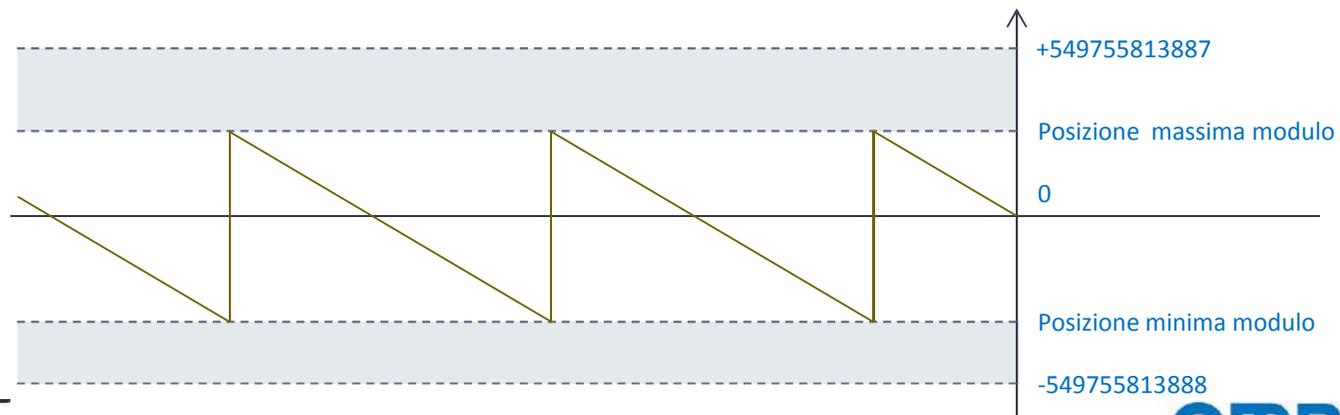
Impostazioni contatore posizione

- La modalità lineare è di norma utilizzata per assi con meccanica con movimenti limitati (es. tavole XY,...)
- La posizione (dopo la conversione in impulsi) è inserita in una variabile integer di 40 bit con segno, quindi può assumere valori da -549755813888 a +549755813887

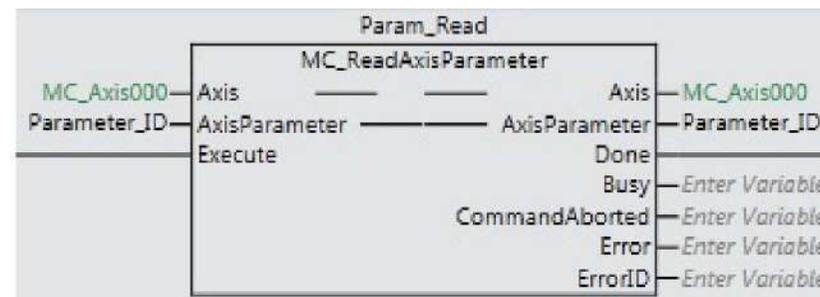
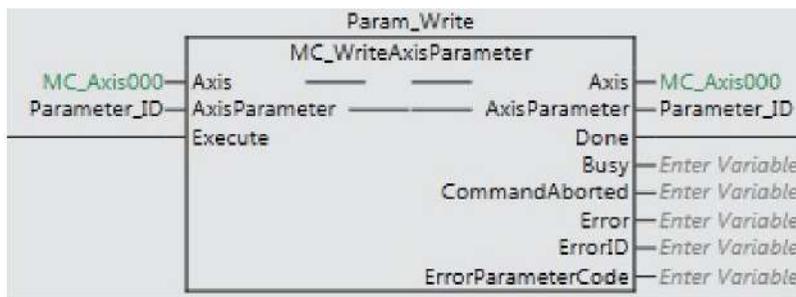


Impostazioni contatore posizione

- La modalità rotativa è invece utilizzata per assi con movimentazioni illimitate (tavole rotanti, avvolgitori,...)
- Il conteggio è inserito di un contatore ad anello di 40 bit ed è quindi effettuato in modo ripetitivo e senza limite
- È necessario specificare i limiti di tale contatore



Scrittura dei parametri dell'asse da programma



NOTE: The value of the parameter rewritten, is not saved in non-volatile memory in the CPU Unit.

How to use:

- 1) Read Axis Parameter list from Axis
- 2) Modify the desired parameters
- 3) Write Axis Parameter list to Axis



Esercizio #1

(Parte 1)

SYSTMAC
always in control

OMRON

Parte 1: Configurazione della rete EtherCAT

- Comporre una rete EtherCAT costituita da un solo nodo slave
- Lo slave deve essere un servozionamento G5 da 100W 200V, con versione firmware 2.1 (codice: R88D-KN01H-ECT Rev:2.1)
- Configurare l'asse nel modulo motion secondo i dati riportati nella slide successiva, chiamandolo 'Asse0'

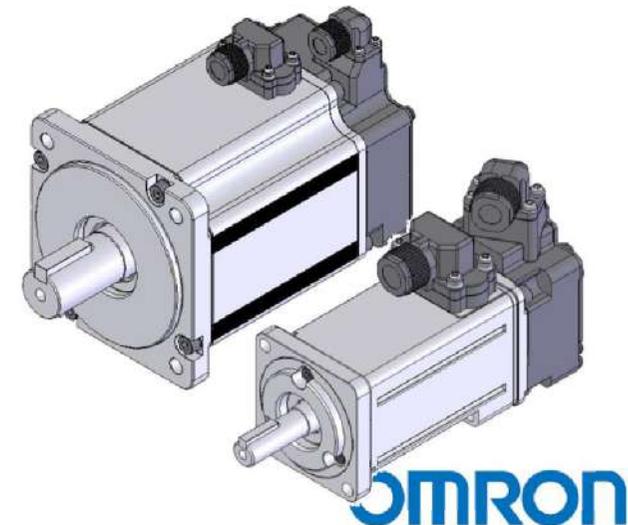


Parte 1: Configurazione degli assi

- **Tipo asse:** Asse servoazionamento
- **Dispositivo d'ingresso:** Nodo 1
- **Contatore impulsi di comando:** 1048576 impulsi/rev
- **Distanza corsa:** 360 °/rev
- **Velocità massima:** 3000 rpm (=18000 °/s)
- **Velocità jog massima:** 1000 rpm (=6000 °/s)
- **Homing:** Posizione zero preimpostata
- **Modalità di conteggio:** rotativa
- **Impostazione modulo:** $\pm 180^\circ$
- **Tipo di encoder:** incrementale

• Soluzione contenuta nel file:

• 'MOTION1 - Istruzioni Motion Discreto.smc'





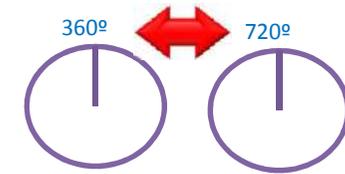
Esercizio #2

SYSTMAC
always in control

OMRON

Scrittura dei parametri dell'asse

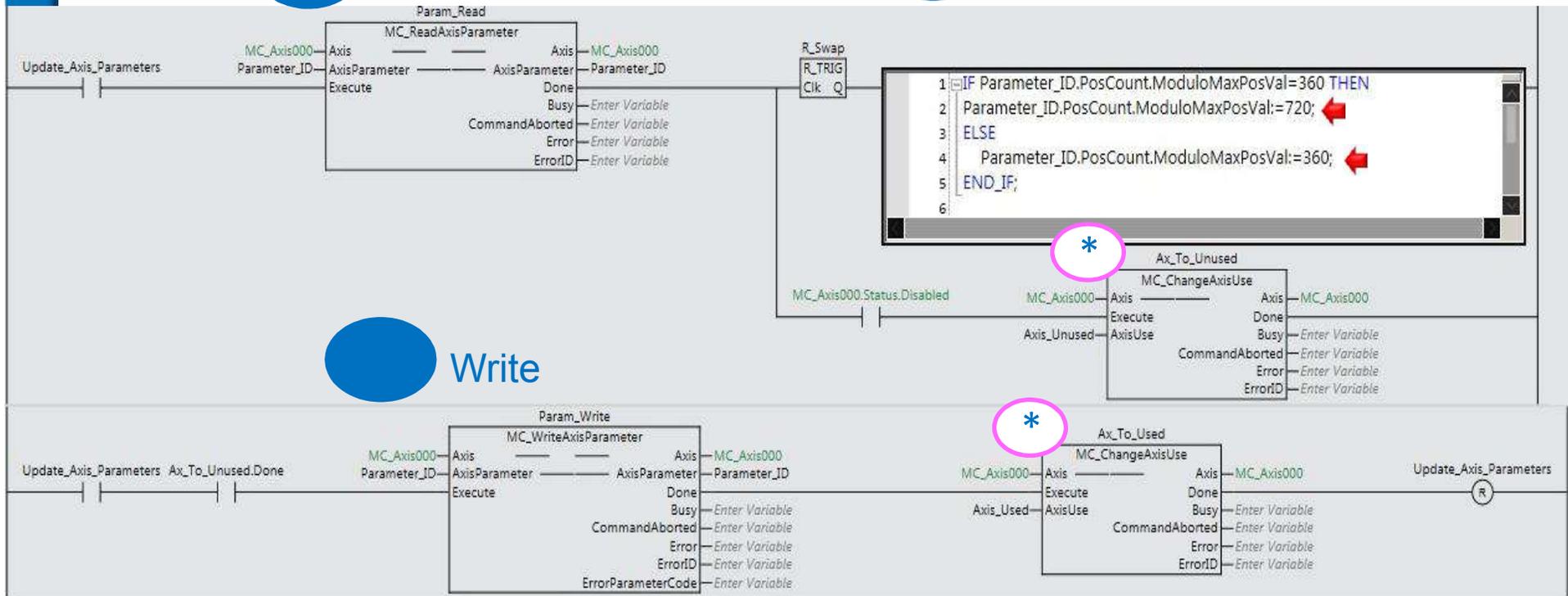
Modificare il modulo di un asse rotativo



Read

Modify

Write





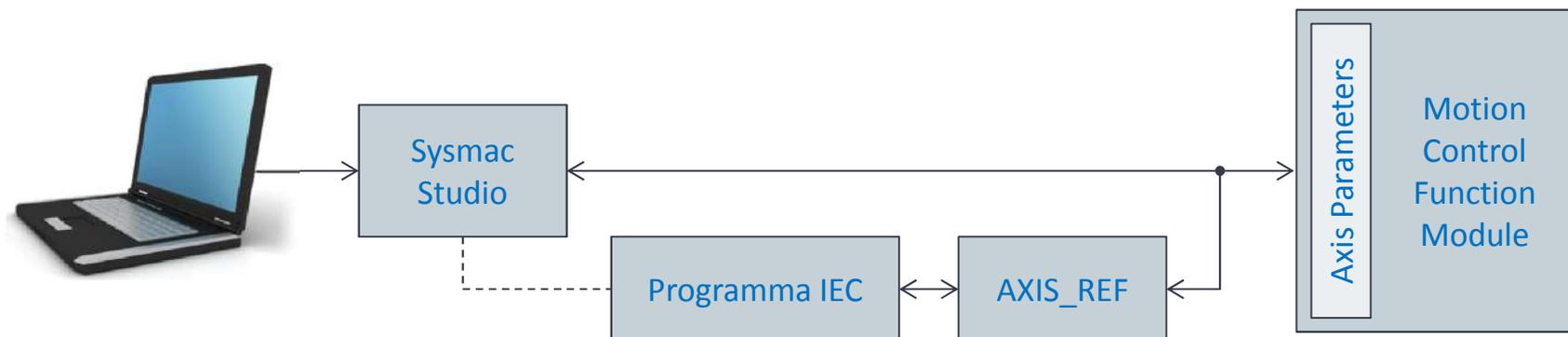
Variabili asse

SYSTMAC
always in control

OMRON

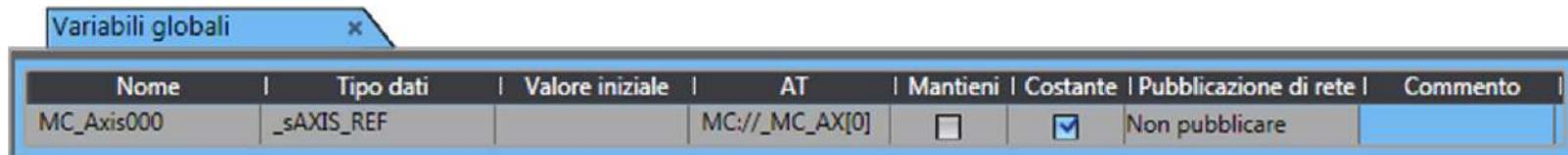
Il tipo di dati `_sAXIS_REF`

- Per il monitoraggio degli assi via programma si utilizza una variabile di tipo `_sAXIS_REF`, che contiene tutti i parametri da monitorare
- Queste variabili fungono da interfaccia sincrona tra modulo MC e programma



Il tipo di dati `_sAXIS_REF`

- Al momento della creazione dell'asse, viene generata una variabile globale (una per ogni asse) che può essere interrogata all'interno del programma
- Il tipo dati `_sAXIS_REF` è una struttura predefinita, che contiene altre strutture. Alcune di esse fanno riferimento a parametri di monitoraggio



Nome	Tipo dati	Valore iniziale	AT	Mantieni	Costante	Pubblicazione di rete	Commento
MC_Axis000	_sAXIS_REF		MC://_MC_AX[0]	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Non pubblicare	

- È possibile far riferimento a questa variabile sia utilizzando il **nome asse**, sia utilizzando l'array `_MC_AX[m]`, dove m indica il <numero dell'asse>

Il tipo di dati `_sAXIS_REF`

Nome	Tipo Dati	Descrizione
<code>.Cfg</code>	<code>_sAXIS_REF_CFG</code>	Configurazione di base dell'asse
<code>.Scale</code>	<code>_sAXIS_REF_SCALE</code>	Rapporto di riduzione elettronica
<code>.Status</code>	<code>_sAXIS_REF_STA</code>	Stato attuale dell'asse
<code>.Details</code>	<code>_sAXIS_REF_DET</code>	Stato di controllo attuale dell'asse
<code>.DrvStatus</code>	<code>_sAXIS_REF_STA_DRV</code>	Stato del drive abbinato all'asse
<code>.Dir</code>	<code>_sAXIS_REF_DIR</code>	Direzione di rotazione dell'asse
<code>.Cmd</code>	<code>_sAXIS_REF_CMD_DATA</code>	Valori di comando del movimento
<code>.Act</code>	<code>_sAXIS_REF_ACT_DATA</code>	Valori di feedback
<code>.Obsr</code>	<code>_sMC_REF_EVENT</code>	Eventuali osservazioni presenti sull'asse
<code>.MFaultLvl</code>	<code>_sMC_REF_EVENT</code>	Eventuali errori non gravi presenti sull'asse

Configurazione di base (CFG)

- I parametri di configurazione disponibili nella `_sAXIS_REF`

Nome	Tipo Dati	Descrizione
<code>.AxNo</code>	UINT	Numero dell'asse
<code>.AxEnable</code>	<code>_eMC_AXIS_USE</code>	Uso dell'asse: 0: <code>_mcNoneAxis</code> (asse non definito) 1: <code>_mcUnusedAxis</code> (asse non utilizzato) 2: <code>_mcUsedAxis</code> (asse utilizzato)
<code>.AxType</code>	<code>_eMC_AXIS_TYPE</code>	Tipo di asse: 0: <code>_mcServo</code> (servoazionamento) 1: <code>_mcEncdr</code> (encoder) 2: <code>_mcVirServo</code> (servoazionamento virtuale) 3: <code>_mcVirEncdr</code> (encoder virtuale)
<code>.NodeAddress</code>	UINT	Indirizzo del nodo EtherCAT abbinato

Rapporto di riduzione elettronica (SCALE)

- I parametri del rapporto di riduzione elettronica presenti in `_sAXIS_REF` sono:

Nome	Tipo Dati	Descrizione
<code>.Num</code>	UDINT	Numeratore del rapporto di riduzione
<code>.Den</code>	LREAL	Denominatore del rapporto di riduzione
<code>.Units</code>	<code>_eMC_UNITS</code>	Modulo di visualizzazione: 0: <code>_mcPls</code> (pulse) 1: <code>_mcMm</code> (mm) 2: <code>_mcUm</code> (μm) 3: <code>_mcNm</code> (nm) 4: <code>_mcDeg</code> (gradi) 5: <code>_mcInch</code> (pollici)

Stato dell'asse (STATUS)

- Indica lo stato PLCopen in cui l'asse opera. E' conseguenza diretta dell'operazione comandata dal modulo MC:

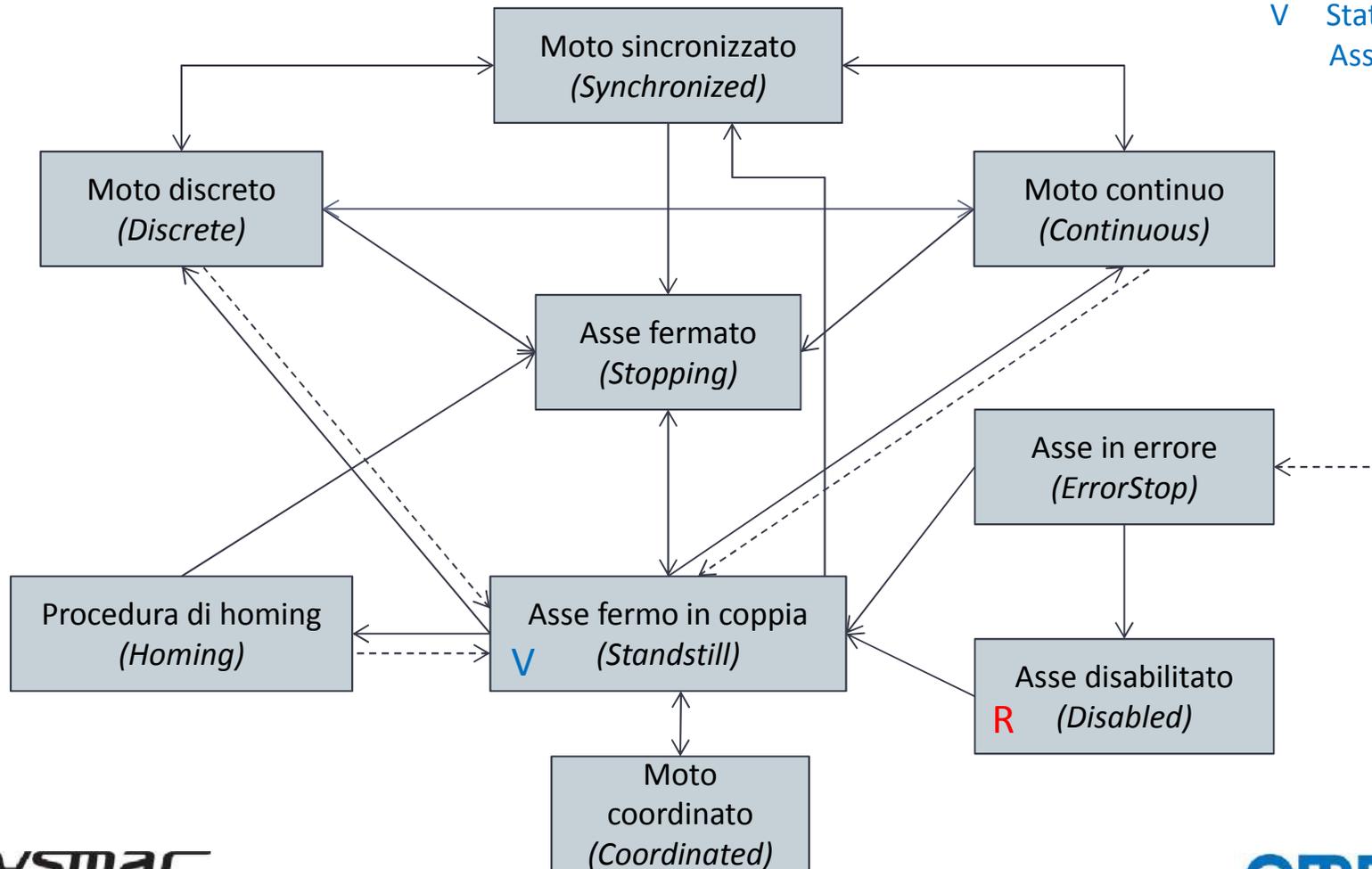
Stato	Descrizione
Disabled	Asse non energizzato
Standstill	Asse energizzato e fermo in coppia
Stopping	Asse fermato con procedura MC_Stop
ErrorStop	Asse in errore o E-STP attivato
Homing	Procedura di homing in esecuzione
Discrete Motion	Posizionamento discreto in esecuzione (movimenti assoluti e relativi...)
Continuous Motion	Posizionamento continuo in esecuzione (controllo di velocità, di coppia, jog...)
Synchronized Motion	Movimento sincronizzato (camme elettroniche, movelink, riduzioni simulate...)
Coordinated Motion	Movimento coordinato (interpolazioni)

Stato dell'asse (STATUS)

- Diagramma di stato PLCopen

R Stato iniziale
Asse Reale

V Stato iniziale
Asse Virtuale



Stato dell'asse (STATUS)

Nome	Tipo Dati	Descrizione
.Ready	BOOL	Asse pronto (in coppia e senza allarmi)
.Disabled	BOOL	Asse non utilizzato
.Standstill	BOOL	Asse fermo in coppia
.Discrete	BOOL	Movimento discreto attivo
.Continuous	BOOL	Movimento continuo attivo
.Synchronized	BOOL	Moto sincronizzato attivo
.Homing	BOOL	Procedura di homing attiva
.Stopping	BOOL	Stop comandato dall'operazione MC_Stop
.ErrorStop	BOOL	Allarme presente o Stop Immediato attivo
.Coordinated	BOOL	Moto coordinato attivo

Stato di controllo (DETAILS)

- I parametri dello stato di controllo inseriti nella struttura `_sAXIS_REF` sono:

Nome	Tipo Dati	Descrizione
<code>.Idle</code>	BOOL	Asse in coppia (in attesa del comando)
<code>.InPosWaiting</code>	BOOL	Asse fermo all'interno del intervallo in-position
<code>.Homed</code>	BOOL	Home dell'asse definito
<code>.InHome</code>	BOOL	L'asse si trova in posizione zero, con l'home definito
<code>.VelLimit</code>	BOOL	Asse in limite di velocità durante il moto sincronizzato

Stato del drive (DRVSTATUS)

- I parametri di stato del drive inseriti nella struttura `_sAXIS_REF` sono:

Nome	Tipo Dati	Descrizione
<code>.ServoOn</code>	BOOL	Asse acceso
<code>.Ready</code>	BOOL	Asse pronto
<code>.MainPower</code>	BOOL	Alimentazione di potenza dell'asse presente
<code>.P_OT</code>	BOOL	Ingresso fine corsa positivo
<code>.N_OT</code>	BOOL	Ingresso fine corsa negativo
<code>.HomeSw</code>	BOOL	Ingresso segnale di prossimità di homing
<code>.Home</code>	BOOL	Ingresso segnale di homing
<code>.ImdStop</code>	BOOL	Ingresso stop immediato

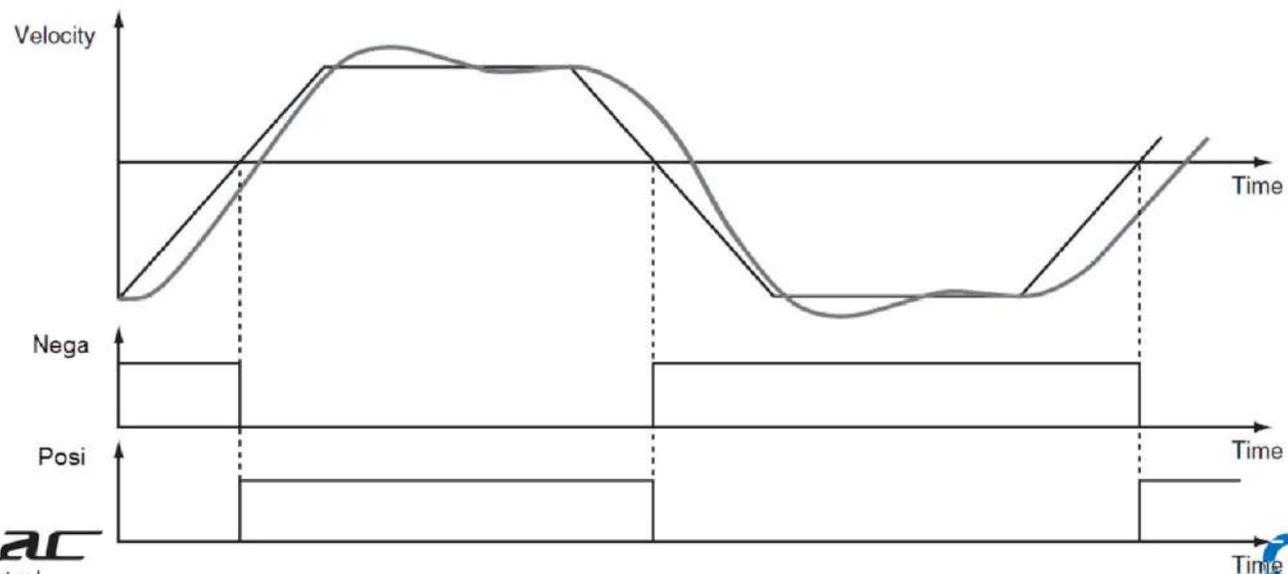
Stato del drive (DRVSTATUS)

Nome	Tipo Dati	Descrizione
.Latch1	BOOL	Ingresso Latch 1
.Latch2	BOOL	Ingresso Latch 2
.DrvAlarm	BOOL	Presente un errore sul drive
.DrvWarning	BOOL	Presente un warning sul drive
.ILA	BOOL	Presente un limite sul drive che influisce sul movimento dell'asse (limiti di coppia, di velocità, limiti software o fine corsa attivi)
.CSP	BOOL	Controllo sincrono di posizione attivo (CSP Mode)
.CSV	BOOL	Controllo sincrono di velocità attivo (CSV Mode)
.CST	BOOL	Controllo sincrono di coppia attivo (CST Mode)

Direzione di rotazione (DIR)

- La direzione di rotazione è indicata tramite le variabili:

Nome	Tipo Dati	Descrizione
.Posi	BOOL	Asse comandato con movimento positivo
.Nega	BOOL	Asse comandato con movimento negativo



Valori di comando (CMD)

- I valori di comando sono inseriti nelle seguenti variabili:

Nome	Tipo Dati	Descrizione
.Pos	LREAL	Comando attuale di posizione (command units)
.Vel	LREAL	Comando attuale di velocità (command units/s)
.AccDec	LREAL	Tasso di accelerazione e di decelerazione del comando attuale (command units/s ²)
.Jerk	LREAL	Tasso di jerk del comando attuale (command units/s ³)
.Trq	LREAL	Comando attuale di coppia (% coppia nominale)

Valori di feedback (ACT)

- I valori del feedback sono inseriti nelle seguenti variabili:

Nome	Tipo Dati	Descrizione
.Pos	LREAL	Feedback attuale di posizione (command units)
.Vel	LREAL	Feedback attuale di velocità (command units/s)
.Trq	LREAL	Feedback attuale di coppia (% coppia nominale)

Osservazioni ed errori non gravi

- Gli errori non gravi(MFAULTLVL) e le osservazioni (OBSR) sono descritte dalle variabili:

Nome	Tipo Dati	Descrizione
.Active	BOOL	Presenza dell'osservazione/errore non grave
.Code	WORD	Codice identificativo dell'osservazione/errore non grave

Esempio

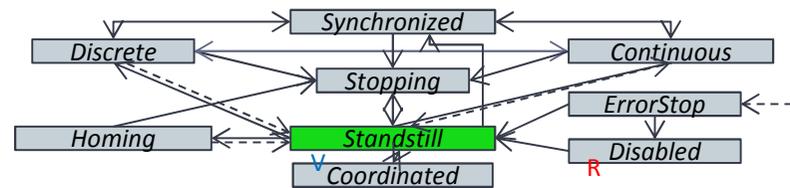
- Dopo aver configurato la rete EtherCAT, impostiamo un asse come:
 - Tipo di asse: servoazionamento
 - Numero asse: 0
 - Indirizzo del nodo: 1
 - Asse in uso

Nome	Valore in linea	Tipo di dati
MC_Axis000.Cfg.AxEnable	2	_eMC_AXIS_USE
MC_Axis000.Cfg.AxNo	0	UINT
MC_Axis000.Cfg.AxType	0	_eMC_AXIS_TYPE
MC_Axis000.Cfg.NodeAddress	1	UINT

Una volta trasferita la configurazione, la variabile `AXIS_REF` ricalcherà quanto precedentemente impostato

Asse Reale

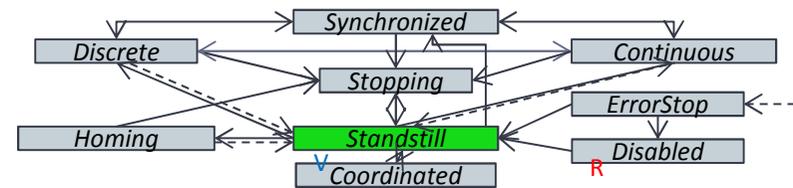
- Inizialmente, all'accensione del sistema, l'asse si trova in stato **Disabled**, in attesa (**Idle**) di istruzione



Nome	Valore in linea	Tipo di dati
MC_Axis000.Status.Ready	True	BOOL
MC_Axis000.Status.Disabled	False	BOOL
MC_Axis000.Status.Standstill	True	BOOL
MC_Axis000.Status.Discrete	False	BOOL
MC_Axis000.Status.Homing	False	BOOL
MC_Axis000.Status.Stopping	False	BOOL
MC_Axis000.Status.ErrorStop	False	BOOL
MC_Axis000.Details.Idle	True	BOOL
MC_Axis000.Details.InPosWaiting	False	BOOL
MC_Axis000.Details.Homed	True	BOOL
MC_Axis000.Details.InHome	True	BOOL

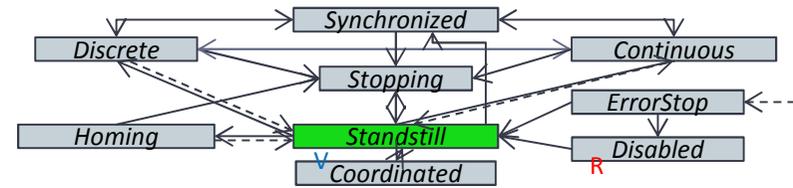
Asse Reale

- Inizialmente, all'accensione del sistema, l'asse si trova in stato **Disabled**, in attesa (**Idle**) di istruzione
- Una volta energizzato, l'asse si pone nello stato **Standstill** e il flag **Ready** viene alzato



Nome	Valore in linea	Tipo di dati
MC_Axis000.Status.Ready	True	BOOL
MC_Axis000.Status.Disabled	False	BOOL
MC_Axis000.Status.Standstill	True	BOOL
MC_Axis000.Status.Discrete	False	BOOL
MC_Axis000.Status.Homing	False	BOOL
MC_Axis000.Status.Stopping	False	BOOL
MC_Axis000.Status.ErrorStop	False	BOOL
MC_Axis000.Details.Idle	True	BOOL
MC_Axis000.Details.InPosWaiting	False	BOOL
MC_Axis000.Details.Homed	True	BOOL
MC_Axis000.Details.InHome	True	BOOL

Asse Reale

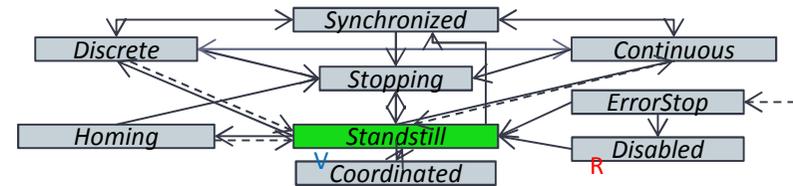


- Inizialmente, all'accensione del sistema, l'asse si trova in stato **Disabled**, in attesa (**Idle**) di istruzione

Nome	Valore in linea	Tipo di dati
MC_Axis000.Status.Ready	True	BOOL
MC_Axis000.Status.Disabled	False	BOOL
MC_Axis000.Status.Standstill	True	BOOL
MC_Axis000.Status.Discrete	False	BOOL
MC_Axis000.Status.Homing	False	BOOL
MC_Axis000.Status.Stopping	False	BOOL
MC_Axis000.Status.ErrorStop	False	BOOL
MC_Axis000.Details.Idle	True	BOOL
MC_Axis000.Details.InPosWaiting	False	BOOL
MC_Axis000.Details.Homed	True	BOOL
MC_Axis000.Details.InHome	True	BOOL

- Una volta energizzato, l'asse si pone nello stato **Standstill** e il flag **Ready** viene alzato
- Quando viene comandata la procedura di homing, l'asse si sposta nello stato **Homing**
- Durante la ricerca dello zero, i flag **Idle** e **Ready** sono bassi

Asse Reale



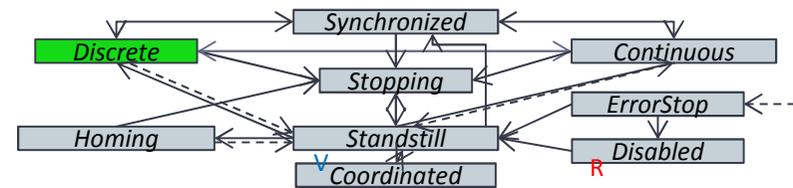
- Una volta terminata la procedura di homing, l'asse torna in attesa di nuove istruzioni (stato **StandStill** e flag **Ready** e **Idle** alti)

Nome	Valore in linea	Tipo di dati
MC_Axis000.Status.Ready	True	BOOL
MC_Axis000.Status.Disabled	False	BOOL
MC_Axis000.Status.Standstill	True	BOOL
MC_Axis000.Status.Discrete	False	BOOL
MC_Axis000.Status.Homing	False	BOOL
MC_Axis000.Status.Stopping	False	BOOL
MC_Axis000.Status.ErrorStop	False	BOOL
MC_Axis000.Details.Idle	True	BOOL
MC_Axis000.Details.InPosWaiting	False	BOOL
MC_Axis000.Details.Homed	True	BOOL
MC_Axis000.Details.InHome	True	BOOL

- Inoltre, il flag **Homed** indica che lo zero dell'asse è stato definito, mentre il flag **InHome** indica che l'asse si trova in posizione zero

Esempio

- Comandando un movimento relativo, l'asse viene posto in stato Moto Discreto (**Discrete**)



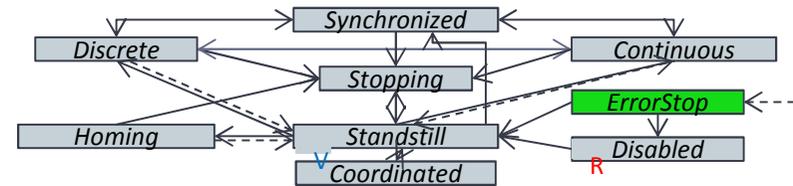
Nome	Valore in linea	Tipo di dati
MC_Axis000.Status.Ready	False	BOOL
MC_Axis000.Status.Disabled	False	BOOL
MC_Axis000.Status.Standstill	False	BOOL
MC_Axis000.Status.Discrete	True	BOOL
MC_Axis000.Status.Homing	False	BOOL
MC_Axis000.Status.Stopping	False	BOOL
MC_Axis000.Status.ErrorStop	False	BOOL
MC_Axis000.Details.Idle	False	BOOL
MC_Axis000.Details.InPosWaiting	False	BOOL
MC_Axis000.Details.Homed	True	BOOL
MC_Axis000.Details.InHome	False	BOOL

Nome	Valore in linea	Tipo di dati
MC_Axis000.Dir.Posi	True	BOOL
MC_Axis000.Cmd.Pos	3767.85	LREAL
MC_Axis000.Cmd.Vel	50	LREAL
MC_Axis000.Cmd.Trq	7.6	LREAL
MC_Axis000.Act.Pos	3766.753196716309	LREAL
MC_Axis000.Act.Vel	49.8199462890625	LREAL
MC_Axis000.Act.Trq	7.6	LREAL

- Le grandezze di comando (**Cmd**) e di feedback (**Act**) varieranno in relazione al movimento imposto

Esempio

- Attivando lo stop immediato (ingresso E-STP), l'asse assumerà lo stato **ErrorStop**, fermando il movimento precedentemente in esecuzione



Nome	Valore in linea	Tipo di dati
MC_Axis000.Status.Ready	False	BOOL
MC_Axis000.Status.Disabled	False	BOOL
MC_Axis000.Status.Standstill	False	BOOL
MC_Axis000.Status.Discrete	False	BOOL
MC_Axis000.Status.Homing	False	BOOL
MC_Axis000.Status.Stopping	False	BOOL
MC_Axis000.Status.ErrorStop	True	BOOL
MC_Axis000.Details.Idle	True	BOOL
MC_Axis000.Details.InPosWaiting	False	BOOL
MC_Axis000.Details.Homed	True	BOOL
MC_Axis000.Details.InHome	False	BOOL

Nome	Valore in linea	Tipo di dati
MC_Axis000.DrvStatus.ImdStop	True	BOOL
MC_Axis000.DrvStatus.Ready	False	BOOL
MC_Axis000.DrvStatus.ServoOn	False	BOOL
MC_Axis000.DrvStatus.DrvAlarm	True	BOOL
MC_Axis000.DrvStatus.DrvWarning	False	BOOL
MC_Axis000.MFaultLvl.Active	True	BOOL
MC_Axis000.MFaultLvl.Code	742F	WORD

- Lo stato del drive mostrerà l'ingresso E-STP (**ImdStop**) e la comparsa di un allarme (**DrvAlarm**), de-energizzando l'asse (**ServoOn**)



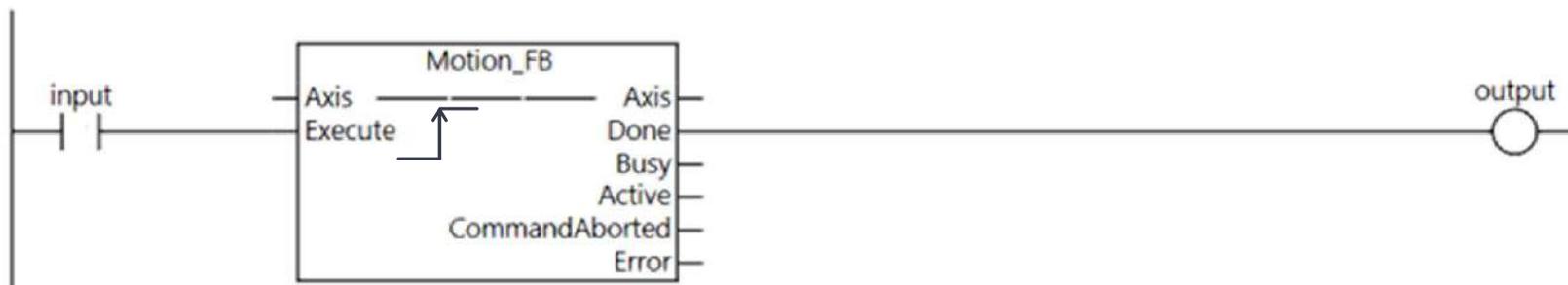
Gestione dei Function Block

SYSTMAC
always in control

OMRON

Struttura di base dei FB di tipo Execute

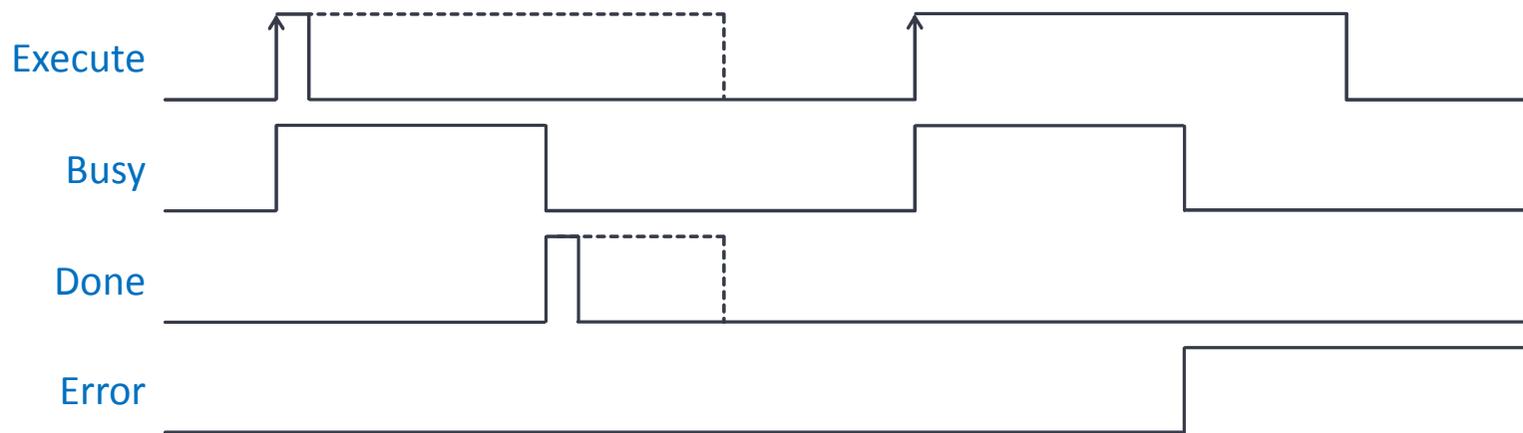
- Il FB detecta internamente il fronte di salita dell'ingresso Execute ed effettua il caricamento delle variabili d'ingresso. Per variare i parametri è necessario dare un secondo fronte di salita all'ingresso Execute
- L'esecuzione prosegue fino a quando:
 - L'operazione non è completata
 - Un'altra operazione non viene eseguita sullo stesso asse
 - Lo stesso FB non viene eseguito nuovamente



Struttura di base dei FB di tipo Execute

- **Done:** si attiva quando l'istruzione termina correttamente o quando la condizione di comando viene raggiunta
- **Active:** si attiva quando l'istanza del FB ottiene il permesso di controllare l'asse a cui è applicata
- **Busy:** si attiva quando l'istanza del FB è in esecuzione
- **CommandAborted:** si attiva quando un'altra istruzione interrompe l'istruzione attualmente in esecuzione
- **Error:** si attiva quando il FB viene eseguito con una variabile fuori dall'intervallo di valori ammissibili

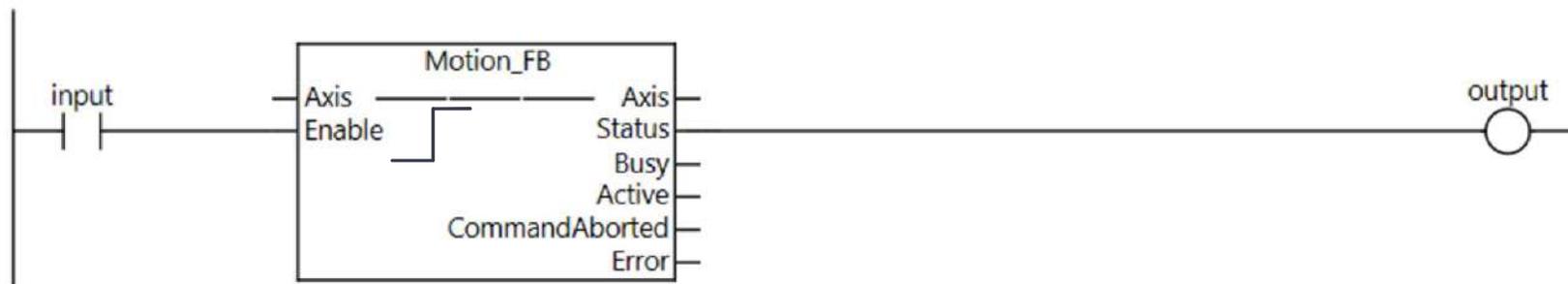
Struttura di base dei FB di tipo Execute



- L'attivazione dei flag Busy, Done, CommandAborted e Error è esclusiva, quindi, in ogni istante, uno solo dei flag sarà attivo
- Si noti che se l'ingresso Execute rimane TRUE per un tempo inferiore al tempo di esecuzione del FB, le uscite Done e CommandAborted rimarranno a TRUE per un solo ciclo di scansione

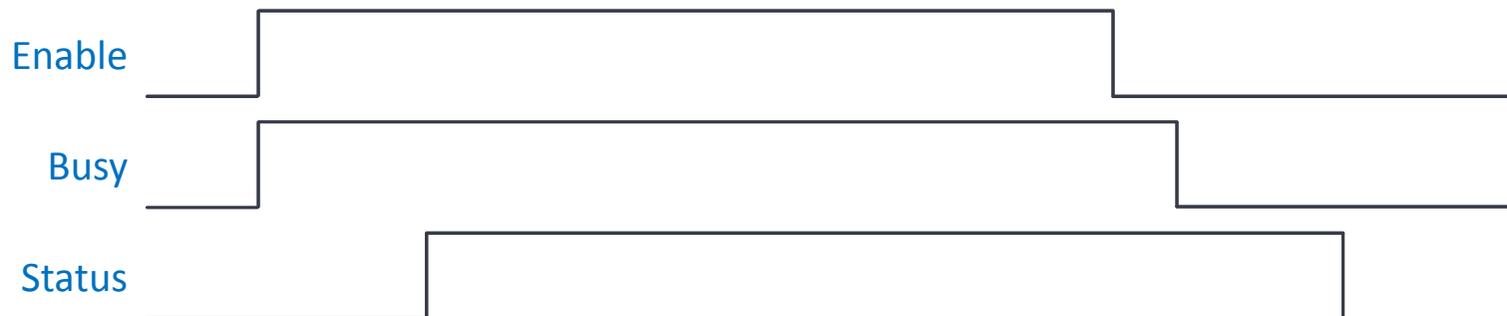
Struttura di base dei FB di tipo Enable

- Tipo **Enable**: il FB viene eseguito ad ogni ciclo, fin tanto che l'ingresso Enable rimane attivo
- Ad ogni ciclo, viene eseguito il caricamento delle variabili d'ingresso che vengono così aggiornate continuamente



Struttura di base dei FB di tipo Enable

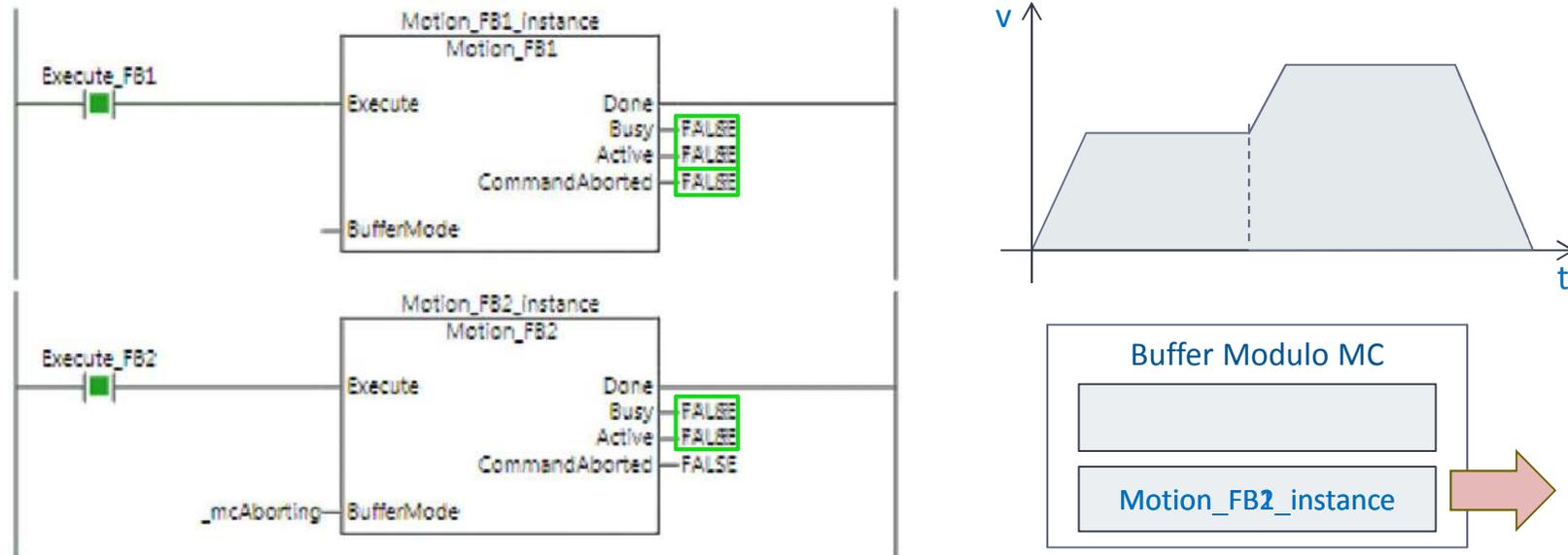
- **Status:** indica che un valore valido ed aggiornato è disponibile sulle uscite del FB
- Busy, CommandAborted ed Error hanno il medesimo significato rispetto ai FB di tipo Execute



Modalità Buffering

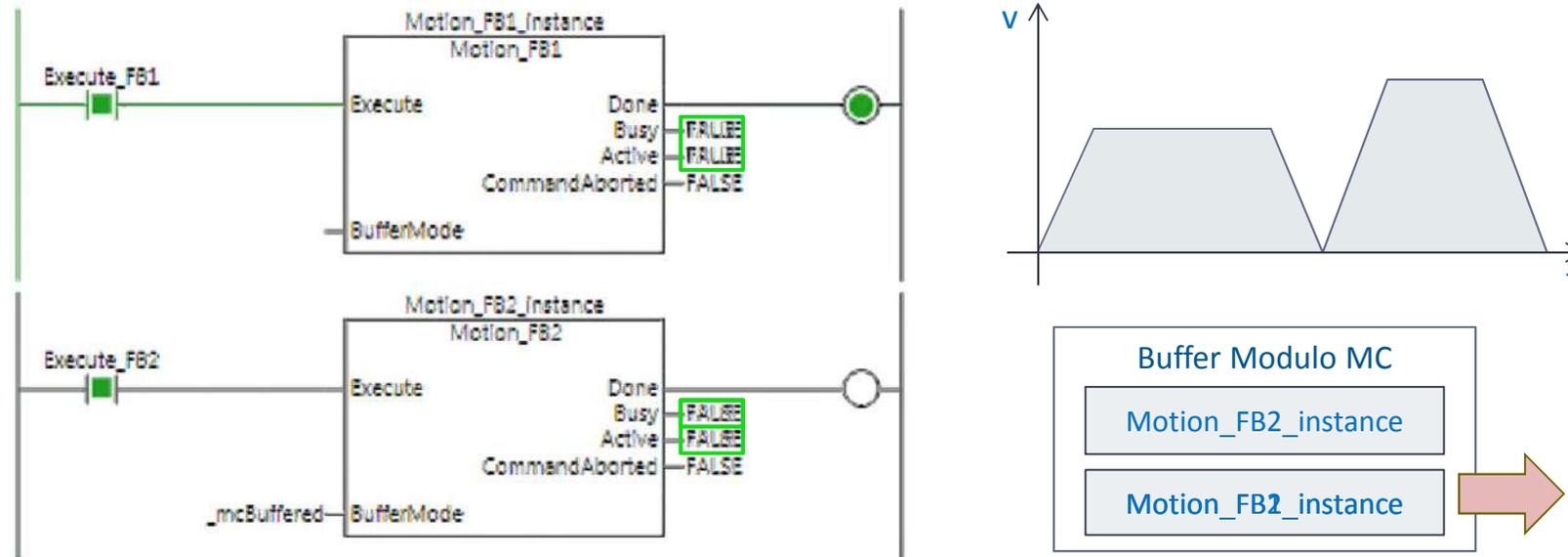
- Il modulo MC è in grado di eseguire una sola Funzione Motion alla volta per ogni asse
- Esiste comunque un buffer con cui si possono accodare istruzioni successive:
 - Per istruzioni a singolo asse, dimensione buffer: 1
 - Per istruzioni con gruppi di assi, dimensione buffer: 7
- L'ingresso BufferMode seleziona la modalità Buffering per ogni FB eseguito. Le opzioni possibili sono:
 - Modalità Aborting
 - Modalità Buffered
 - Modalità Blending

Modalità Buffering - Aborting



- La modalità aborting ferma immediatamente l'esecuzione del FB in fase di processo ed attiva l'esecuzione del successivo. Sul primo FB, l'arresto nell'esecuzione è segnalato mediante l'uscita `CommandAborted`, mentre nel buffer del modulo MC, la prima istanza è sostituita dalla seconda.

Modalità Buffering - Buffering



- La modalità buffering permette alla seconda operazione di essere memorizzata nel buffer e di venire attivata automaticamente al termine della prima.

Modalità Buffering - Blending

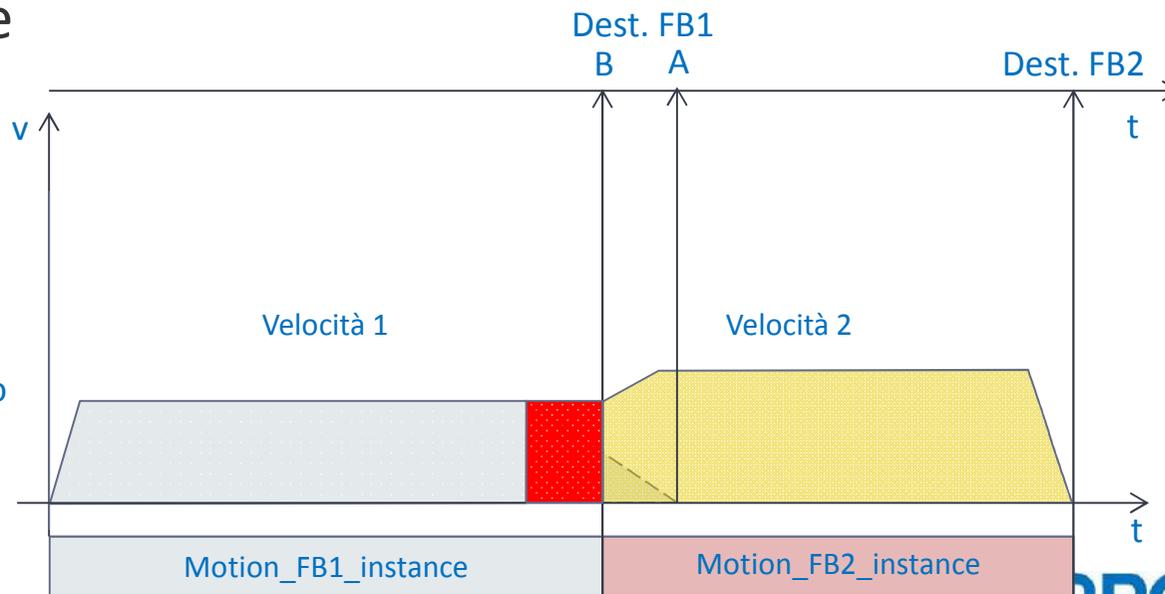
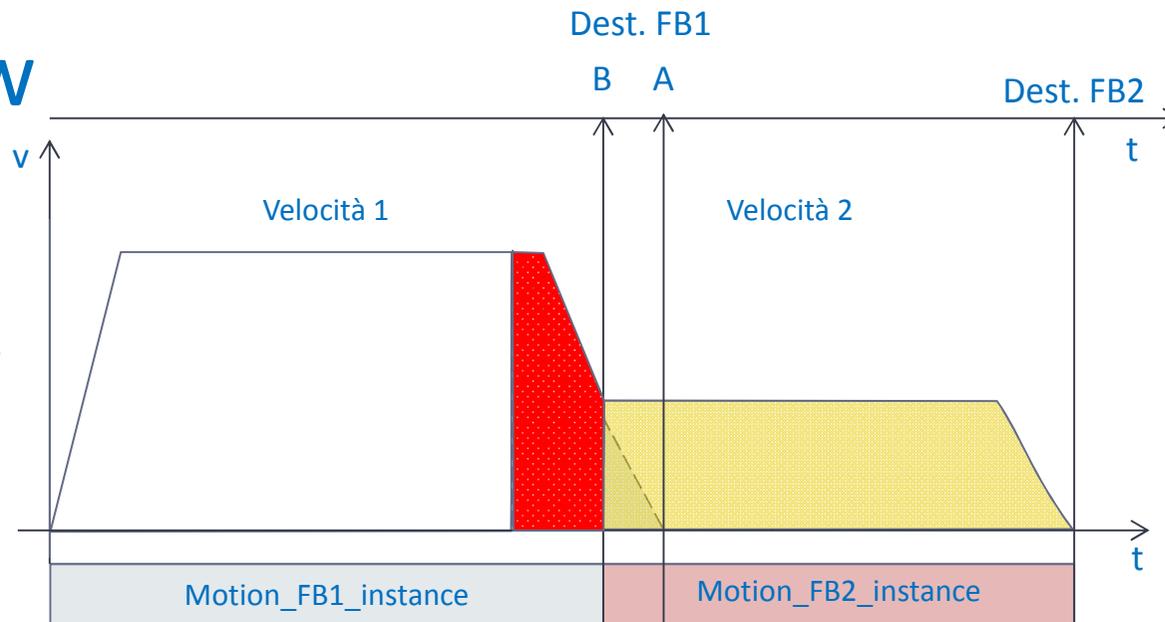
- La modalità blending calcola la velocità punto per punto, necessaria per passare da un comando Motion al successivo
- La modalità fa uso del buffer di istruzioni del Modulo MC
- Il passaggio tra i due comandi avviene, così, in modo morbido
- Esistono quattro tipi di Blending Mode:
 - Blending Low
 - Blending High
 - Blending Previous
 - Blending Next

Nome	Tipo dati	Valore iniziale	AT	Mantieni	Costante	Commento
BlendingMode	_eMC_BUFFER_MODE			<input type="checkbox"/>	<input type="checkbox"/>	

- _mcAborting
- _mcBuffered
- _mcBlendingLow
- _mcBlendingPrevious
- _mcBlendingNext
- _mcBlendingHigh

Blending Low

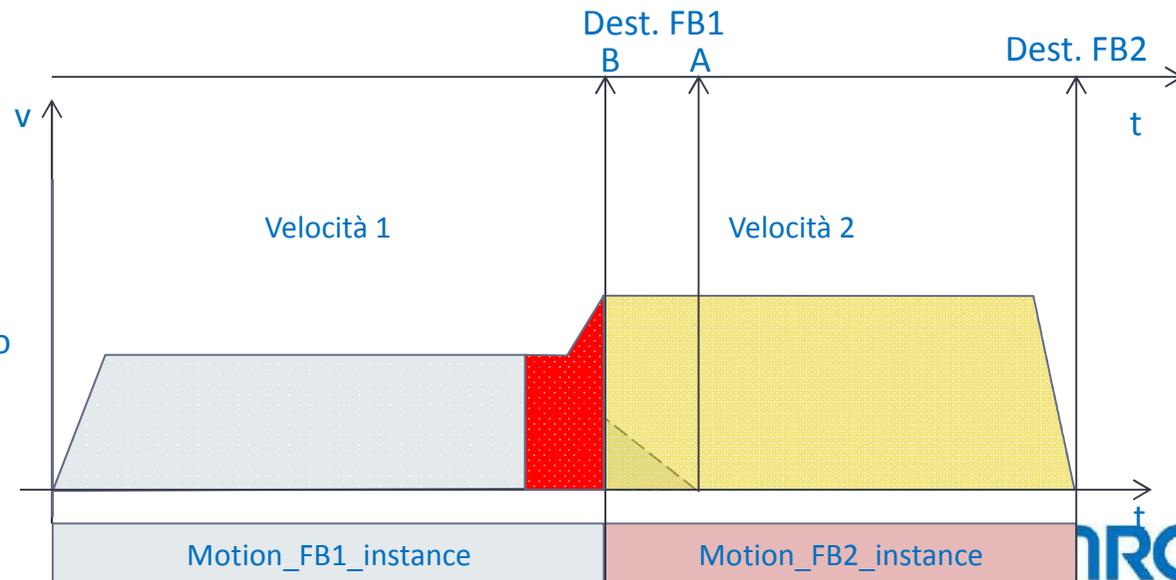
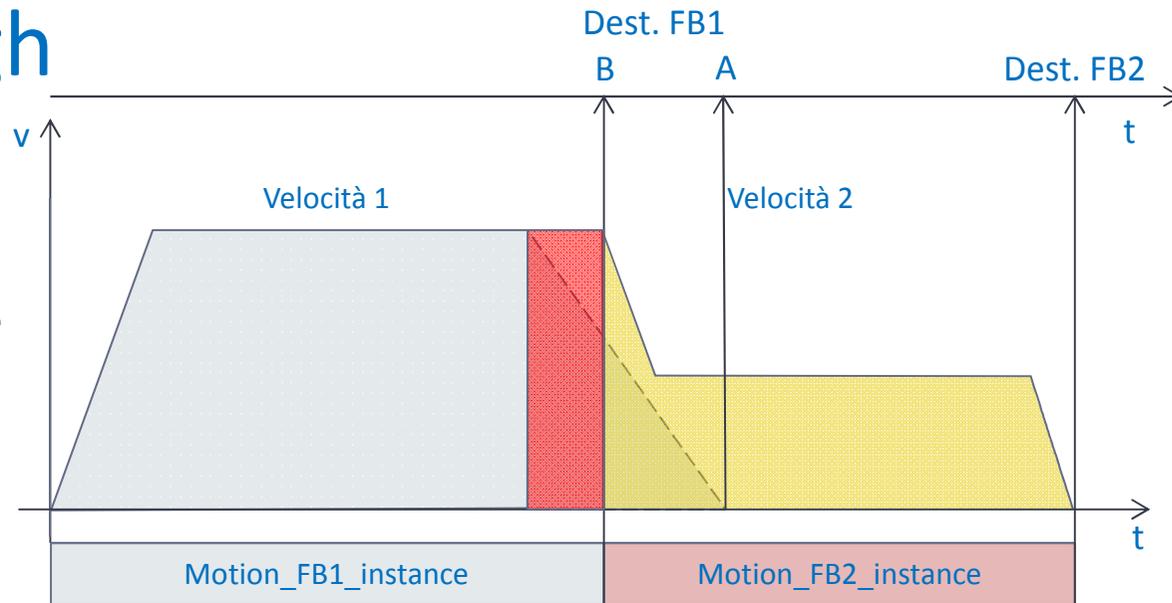
- Il collegamento tra le istruzioni avviene alla velocità più bassa fra quelle di comando dell'istruzione FB1 e FB2



Dest. FB1: Tempo finale movimento
A. senza Blending
B. con Blending

Blending High

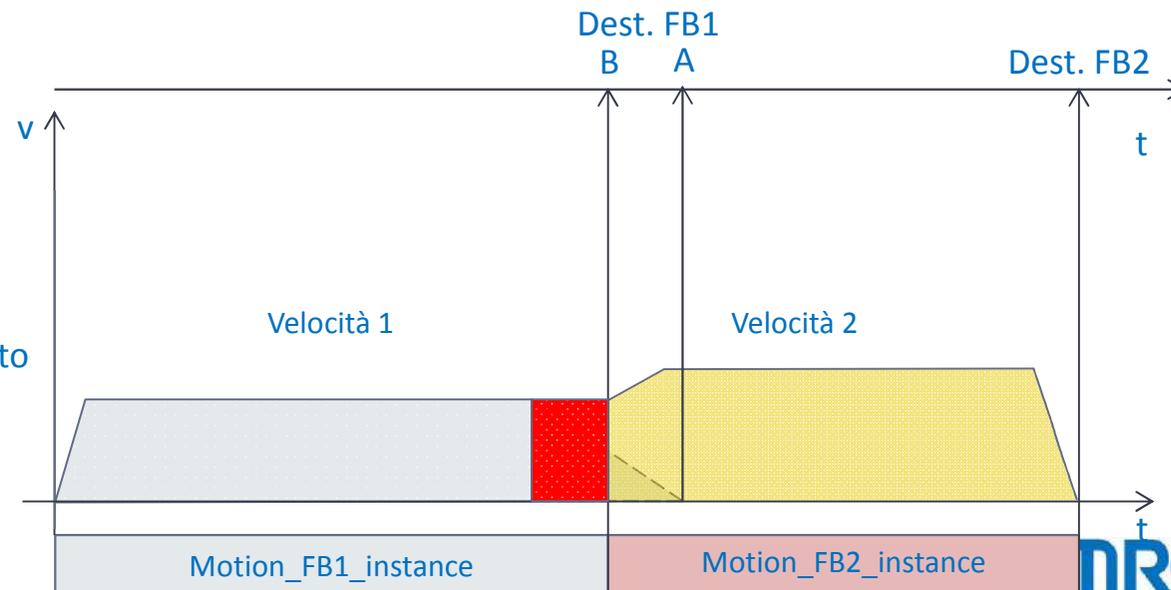
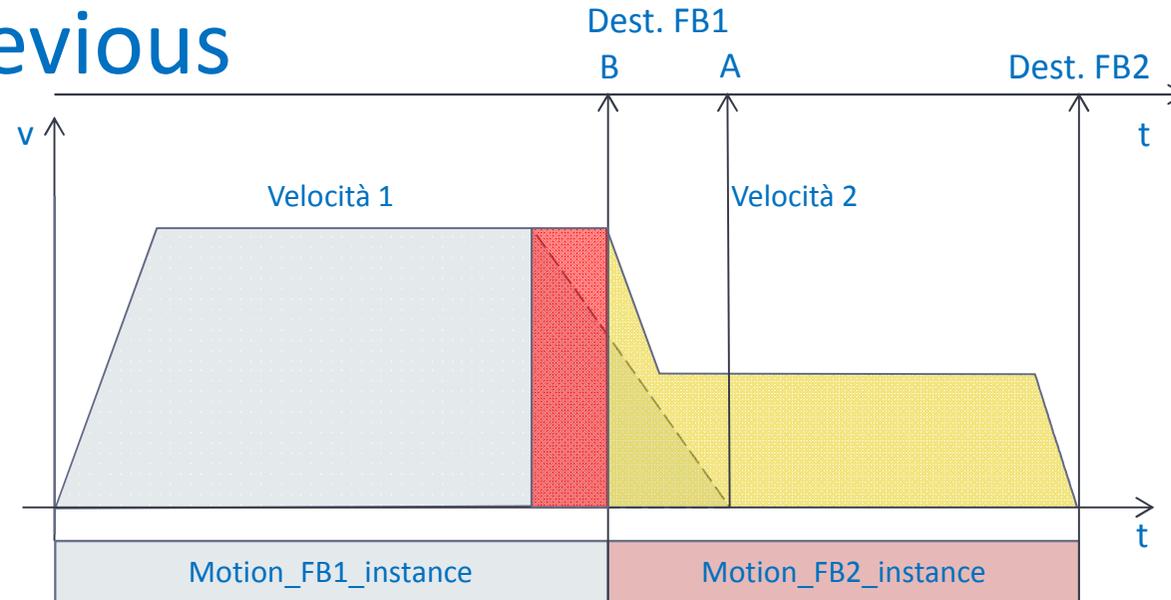
- Il collegamento tra le istruzioni avviene alla velocità più alta fra quelle di comando dell'istruzione FB1 e FB2



Dest. FB1: Tempo finale movimento
 A. senza Blending
 B. con Blending

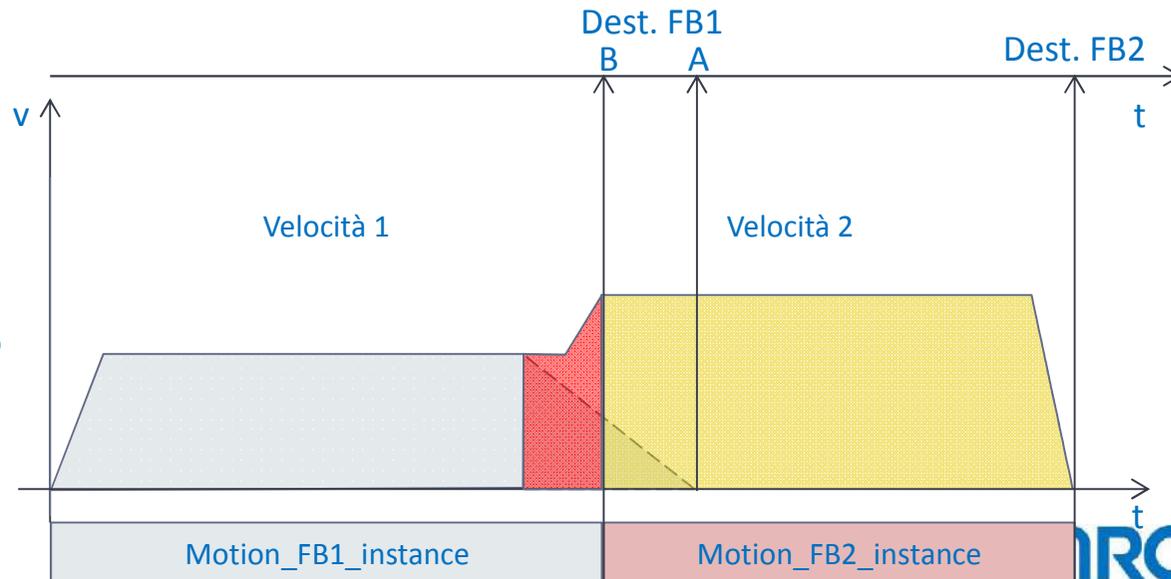
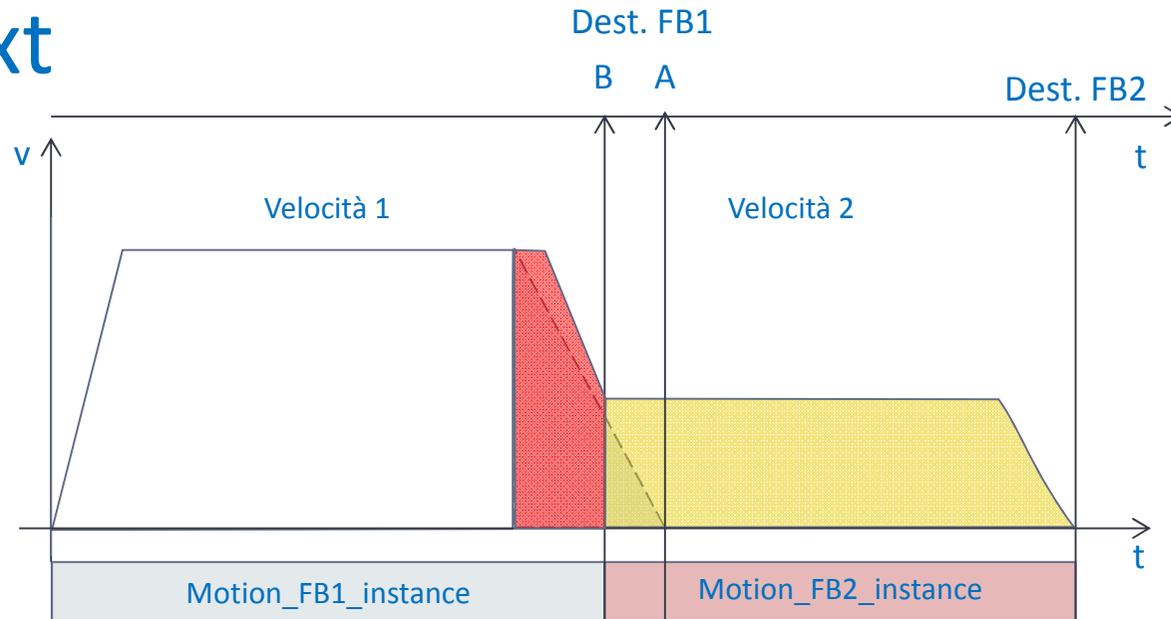
Blending Previous

- Il collegamento tra le istruzioni di posizione avviene alla velocità di comando dell'istruzione FB10



Blending Next

- Il collegamento tra le istruzioni di posizione avviene alla velocità di comando dell'istruzione FB2



Dest. FB1: Tempo finale movimento
A. senza Blending
B. con Blending



Domande?

E OY!



SYSTMAC
always in control

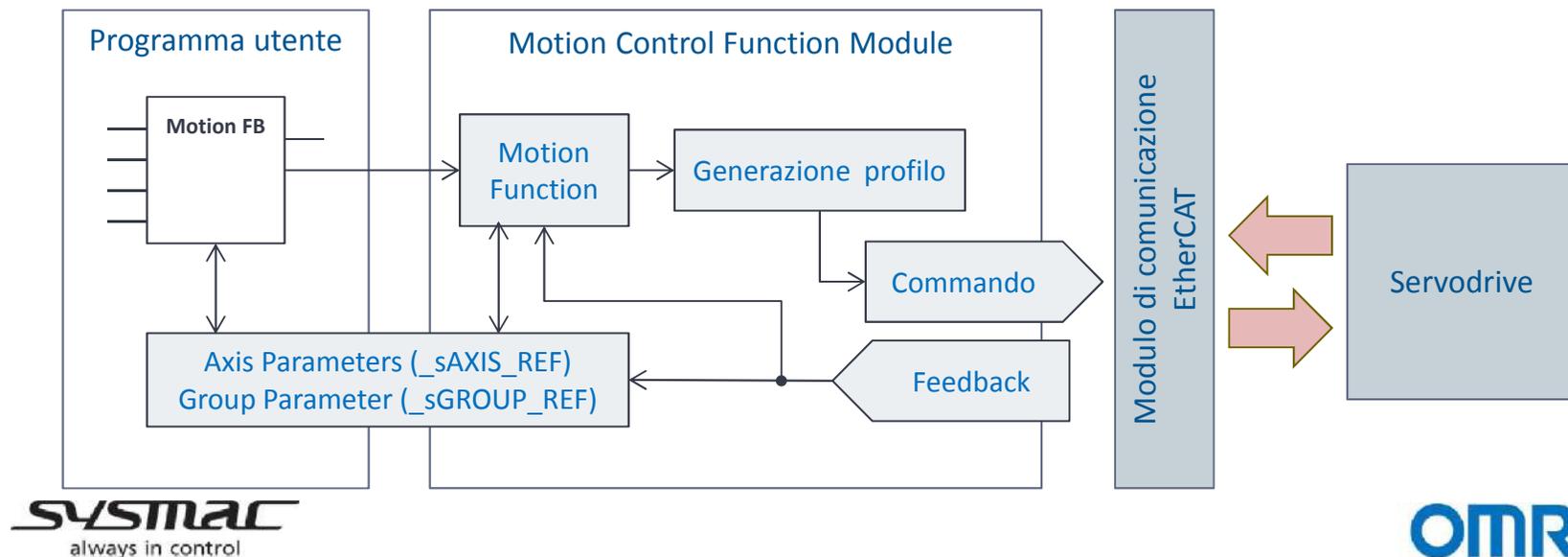
OMRON



Comandi Motion Discreti

Funzioni Motion Control

- Le **Funzioni Motion Control**, implementate nel Modulo MC, sono dedicate al controllo degli assi
- I **Function Block Motion Control**, richiamati dal programma IEC, posseggono la certificazione IEC 61131-3 e richiamano univocamente le Funzioni Motion Control



Funzioni Motion Control

- Le Funzioni Motion Control si possono classificare in 2 categorie:

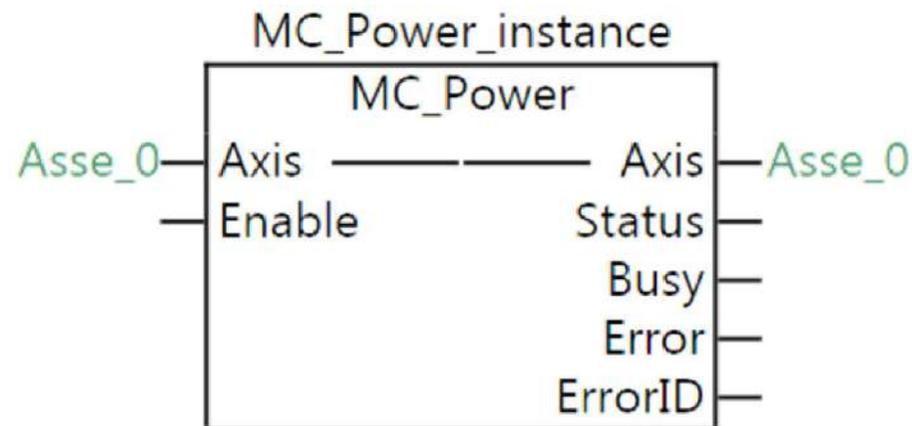
Comandi Motion	Comandi Amministrativi
Questi comandi generano variazioni nei profili di posizione, velocità o coppia. Es: MoveAbsolute, MoveRelative, CamIn, MoveVelocity, Torque, ...	Questi comandi non generano direttamente una variazione nel profilo. Es: Power, Home, SetPosition, TouchProbe, ZoneSwitch

La categoria Comandi Motion si può dividere in 2 sottocategorie:

Comandi ad asse singolo	Comandi per gruppo assi
Questi comandi influenzano il comportamento di un singolo asse	Questi comandi influenzano il comportamento di un gruppo di più assi (interpolazioni, movimenti coordinati, ...)

MC_Power

- La funzione MC_Power serve per mettere in coppia l'asse:

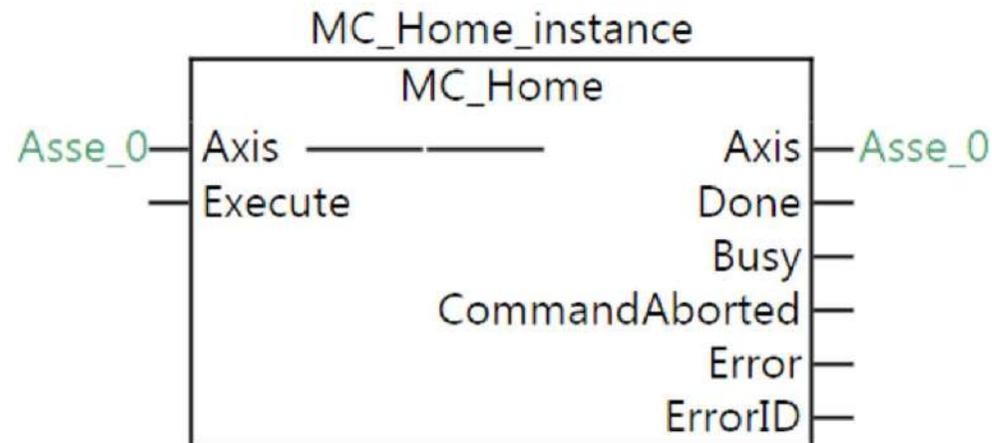


Quando l'ingresso Enable diventa TRUE, l'asse viene messo in coppia ed è possibile effettuare il controllo. L'uscita Status è indice del reale stato dell'asse.

È possibile utilizzare questo comando solo per assi servo. Gli assi virtuali sono già abilitati per definizione. Se dovesse venire utilizzato per altri tipi di assi, viene generato un errore.

MC_Home

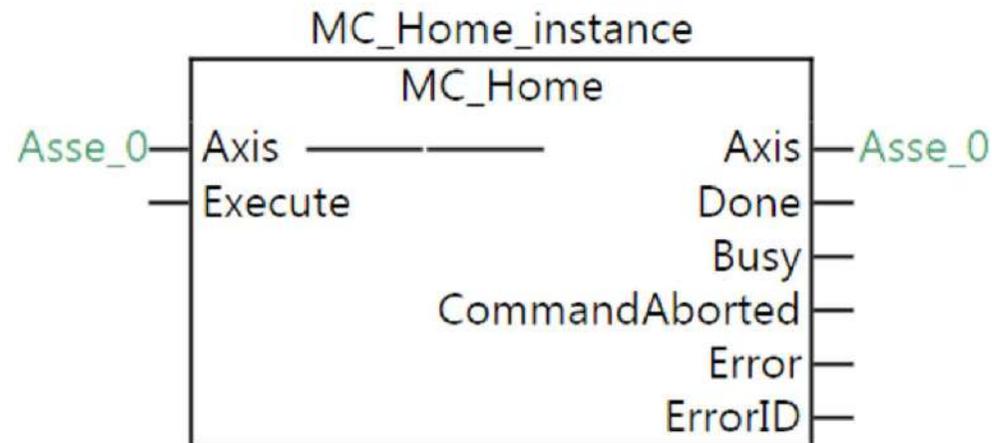
- La funzione MC_Home serve per effettuare la procedura di homing con parametri impostati nella configurazione dell'asse:



Sul fronte di salita del segnale Execute, la procedura di homing verrà abilitata secondo le specifiche impostate. Durante l'esecuzione dell'homing, verrà attivato il flag **Status.Homing** relativo all'asse, mentre al termine verrà attivato il flag **Details.Homed**.

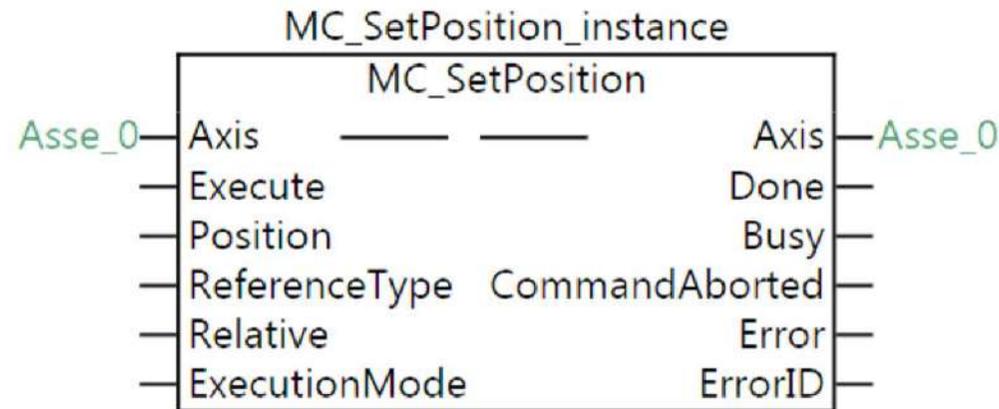
MC_HomeWithParameter

- La funzione MC_HomeWithParameter serve per effettuare la procedura di homing con parametri impostati dall'utente:



Sul fronte di salita del segnale Execute, la procedura di homing verrà abilitata secondo le specifiche impostate. Durante l'esecuzione dell'homing, verrà attivato il flag **Status.Homing** relativo all'asse, mentre al termine verrà attivato il flag **Details.Homed**.

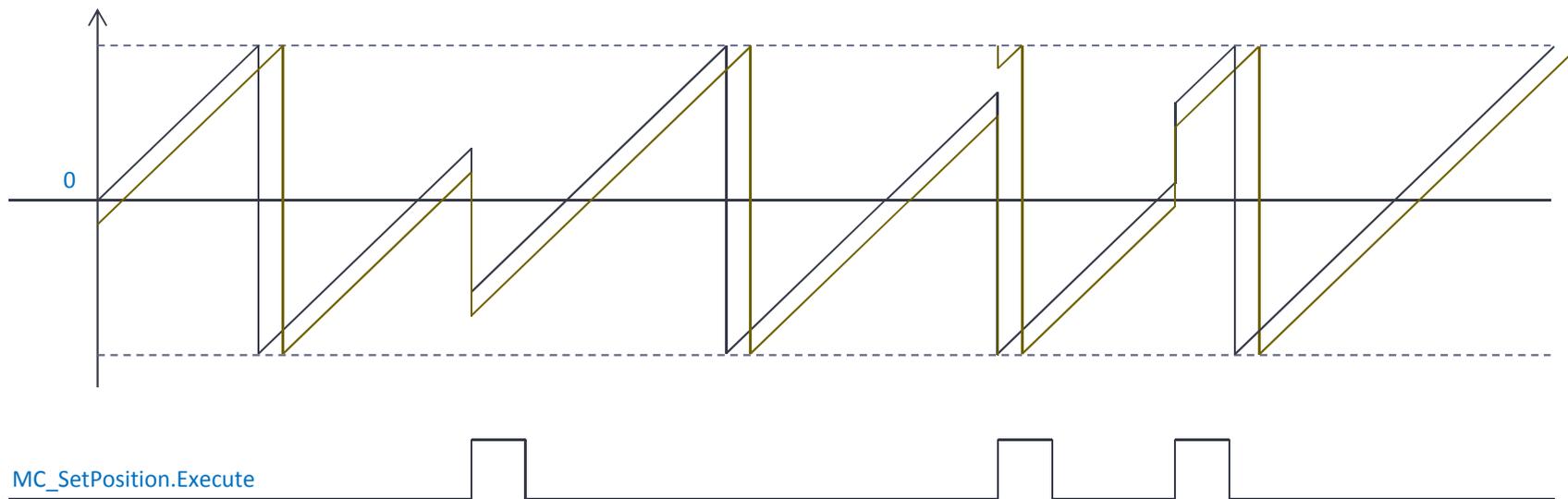
MC_SetPosition



- La funzione SetPosition ridefinisce la posizione dell'asse con un nuovo valore di posizione:
 - **Position** identifica la nuova posizione con cui ridefinire l'asse
 - **ReferenceType**, invece, indica se la ridefinizione deve essere effettuata sui valori del feedback o del comando
 - **ExecutionMode** e **Relative** sono ingressi riservati a future implementazioni

MC_SetPosition

- Quando viene eseguita l'istruzione MC_SetPosition, i valori di comando e di feedback vengono aggiornati nello stesso istante, in modo da mantenere costante il following error prima e dopo la transizione.

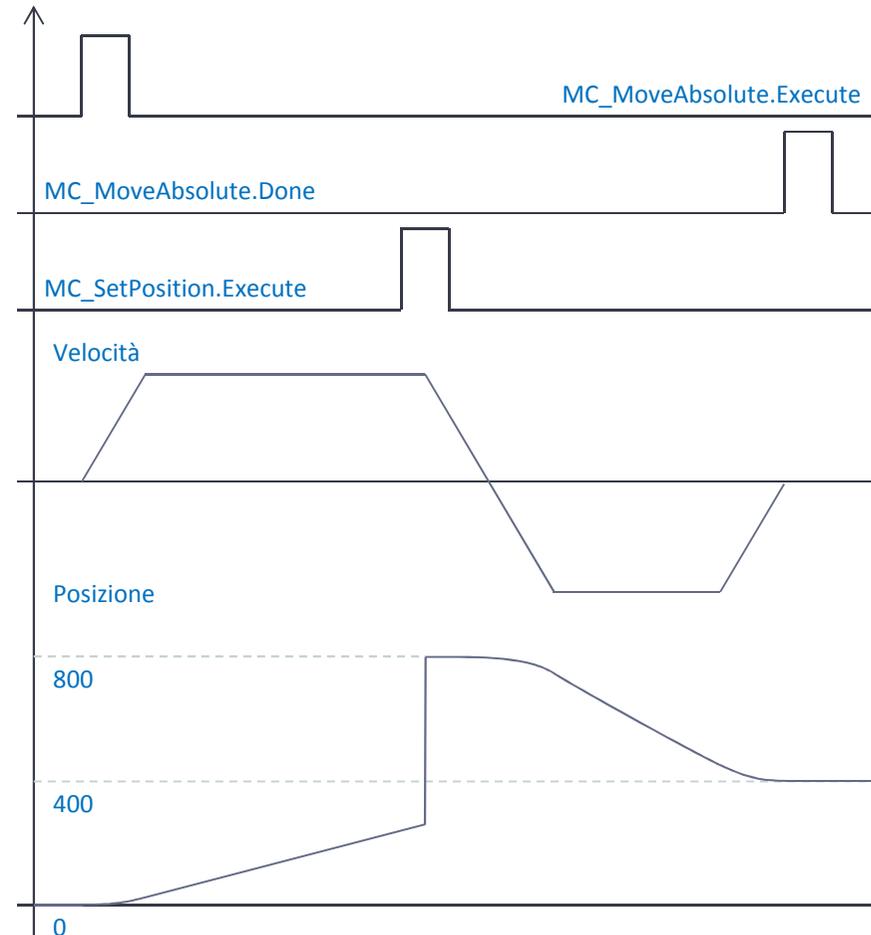


MC_SetPosition

- Porre attenzione quando si utilizza l'istruzione MC_SetPosition in corrispondenza di un movimento assoluto: variando la posizione dell'asse, il target viene mantenuto costante, quindi si potrebbero verificare dei movimenti indesiderati.

- Esempio.**

- Movimento assoluto da 0 a 400.
- Posizione prima di SetPosition = 200
- Posizione dopo SetPosition = 800





Esercizio #1

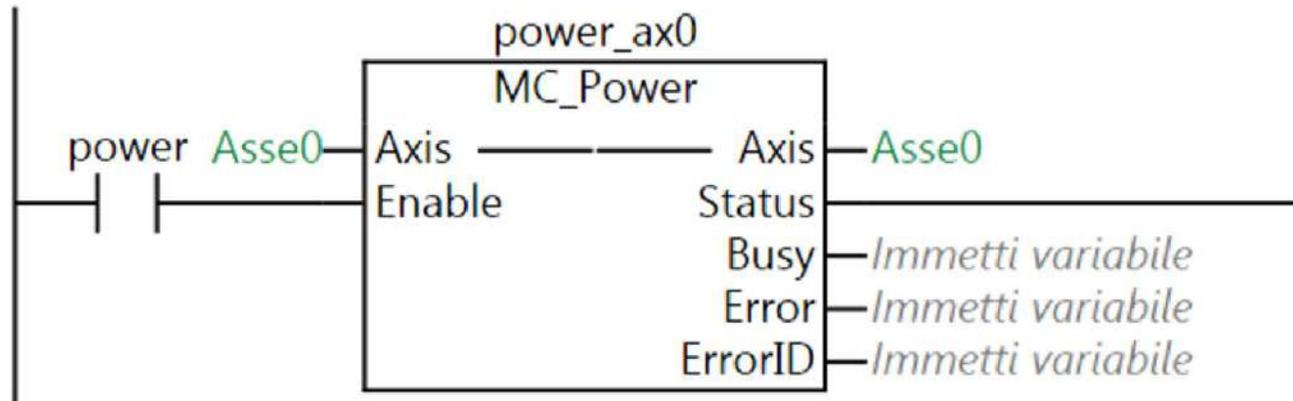
(Parte 2)

SYSTMAC
always in control

OMRON

Parte 2: Messa in coppia

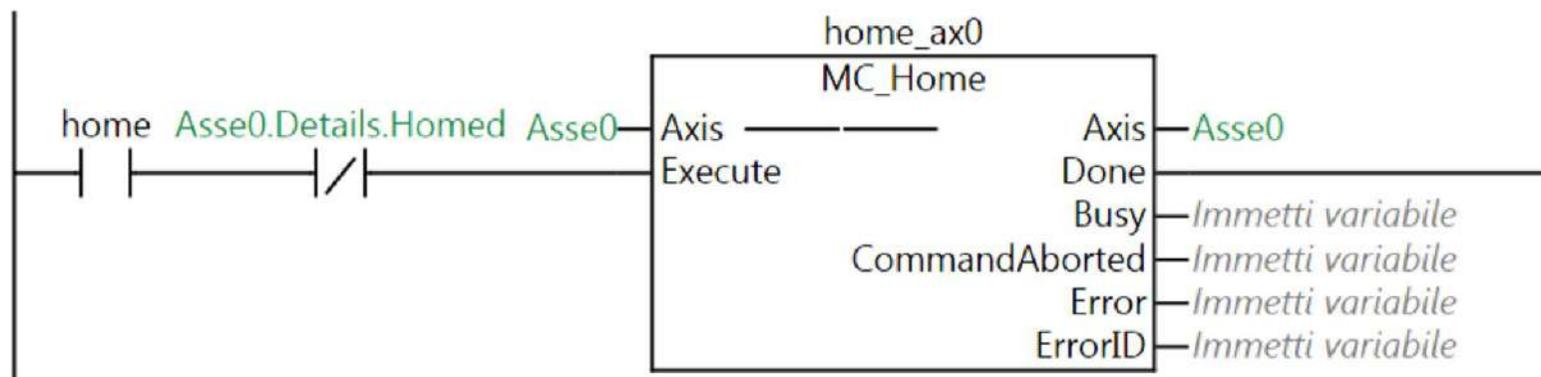
- Utilizzare il Function Block MC_Power per mettere l'asse in coppia.



- Controlla lo stato dell'asse, tramite la variabile di sistema AXIS_REF (Asse0.Status e Asse0.Details).

Parte 2: ricerca dello zero

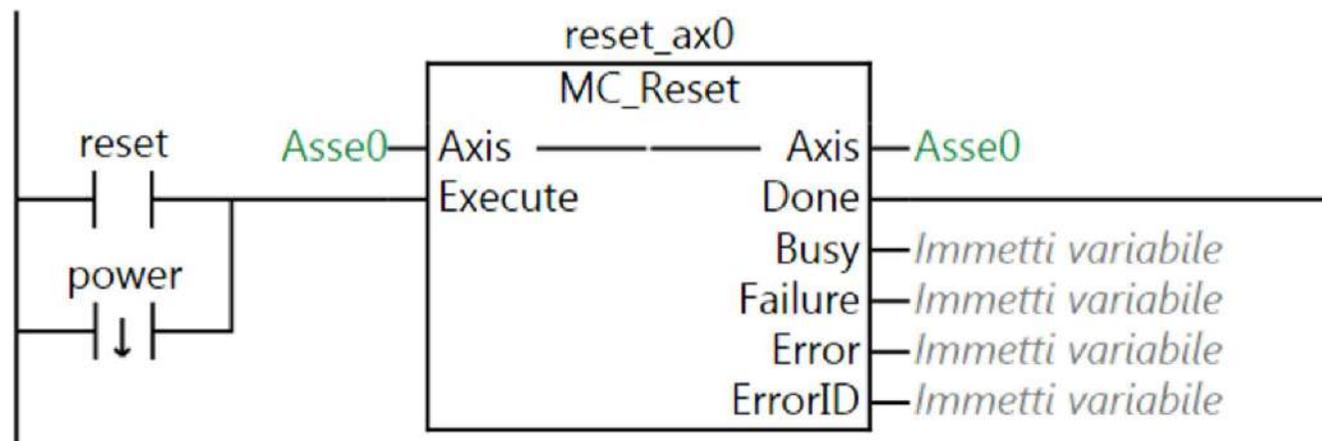
- Utilizzare il Function Block MC_Home per effettuare la procedura di ricerca dello zero precedentemente impostata.



- Asse0.Details.Homed** indica se la ricerca dello zero è già stata effettuata

Parte 2: reset degli errori dell'asse

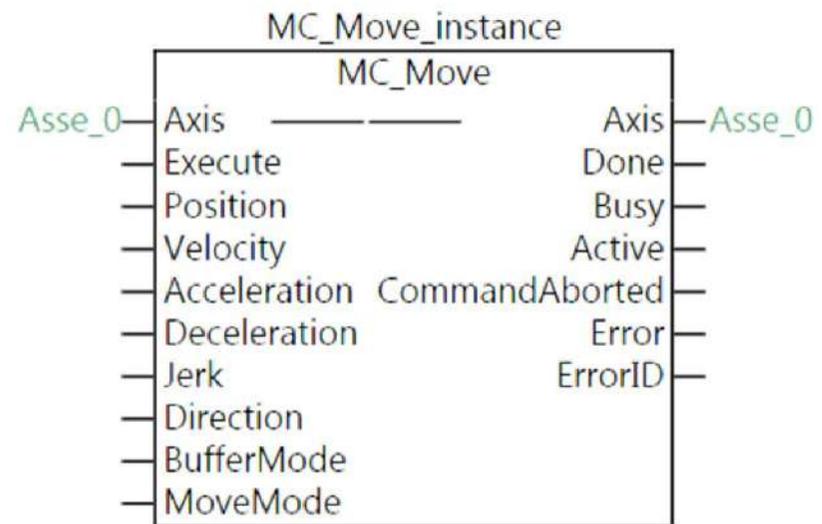
- Utilizzare il Function Block MC_Reset per cancellare eventuali errori presenti sull'asse e/o sul servodrive.



- Il reset viene eseguito con il contatto «reset» o quando l'asse viene spento.

MC_Move

- L'istruzione di movimento più semplice è la MC_Move, che opera una movimentazione a singolo asse (assoluta o relativa).
- Nella variabile In-Out del FB è necessario inserire l'asse (che deve essere abilitato e pronto) sul quale effettuare l'operazione.
- Le specifiche del movimento (espresse in unità) devono essere indicate negli ingressi **Position**, **Acceleration**, **Deceleration** e **Jerk**. L'ingresso **MoveMode** specifica la natura assoluta o relativa del movimento.

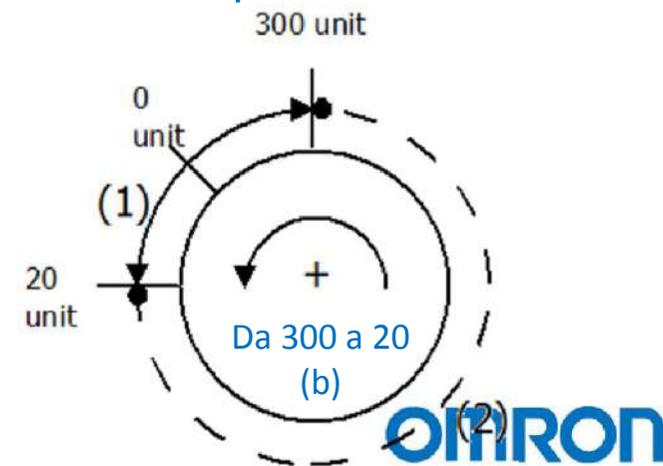
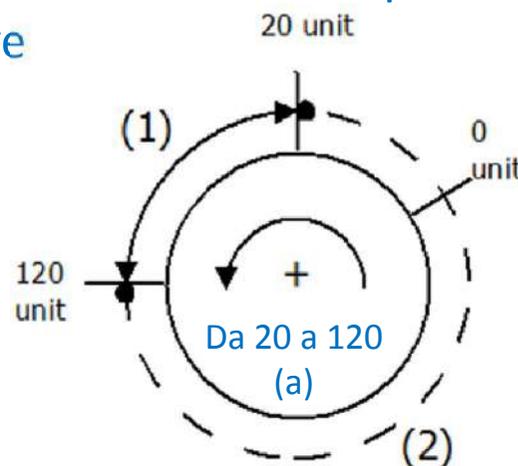


MC_Move

• L'ingresso **Direction** è utilizzato solamente nel caso di movimenti assoluti su assi in modalità rotativa:

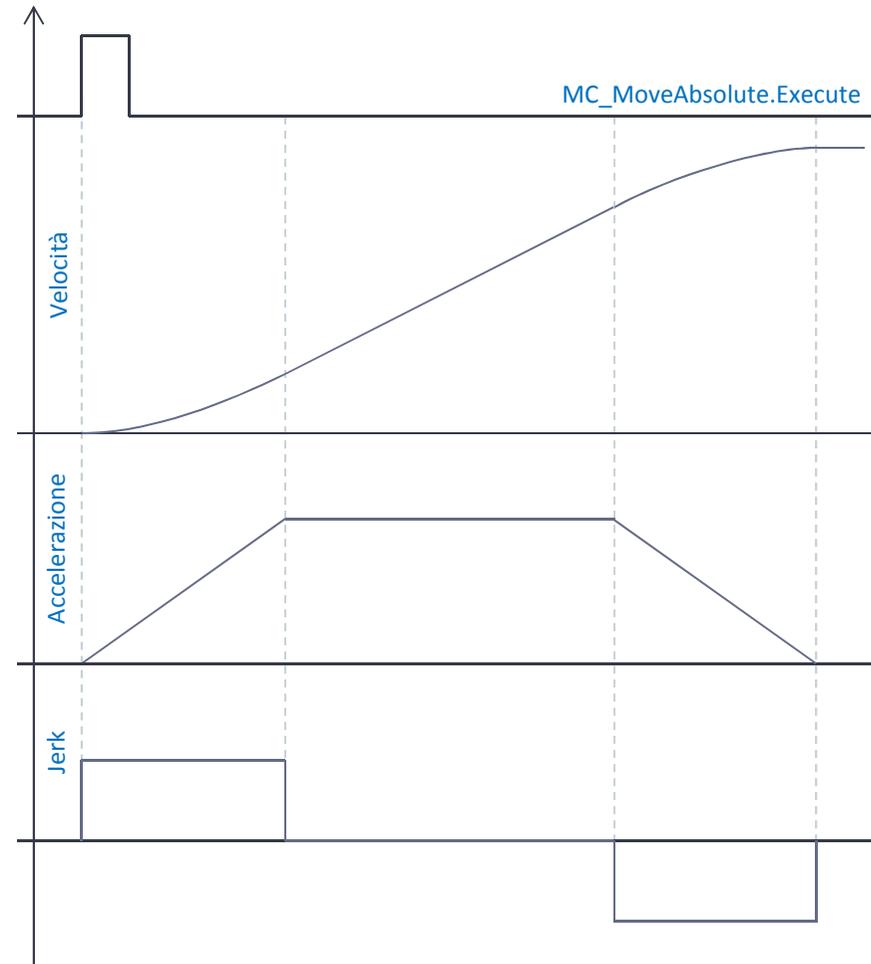
- **_mcPositiveDirection**: la direzione di movimento è in avanti (a)(1)/(b)(1)
- **_mcNegativeDirection**: la direzione di movimento è all'indietro (a)(2)/(b)(2)
- **_mcShortestWay**: la direzione è quella che permette il movimento più breve (a)(1)/(b)(1)
- **_mcNoDirection**: la direzione è quella che permette il movimento all'interno dello stesso Ring Counter (a)(1)/(b)(2)
- **_mcCurrentDirection**: la direzione è quella indicata nella precedente istruzione MC_Move

Asse rotativo modulo 0 - 360

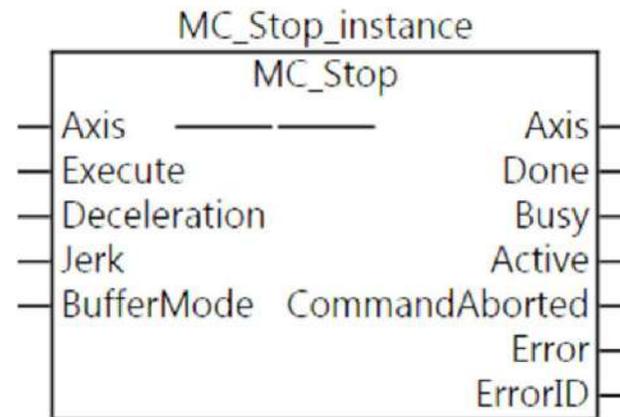


MC_Move

- L'ingresso di **Jerk** pone un limite alla pendenza massima di variazione dell'accelerazione.
- Il profilo trapezoidale dell'accelerazione produce un andamento più morbido della velocità, in modo da prevenire eventuali strappi sul carico.



MC_Stop



La funzione Stop annulla il movimento in corso, fermando l'asse in decelerazione, secondo il valore scritto nelle variabili Deceleration e Jerk.



Esercizio #1

(Parte 3)

SYSTMAC
always in control

OMRON

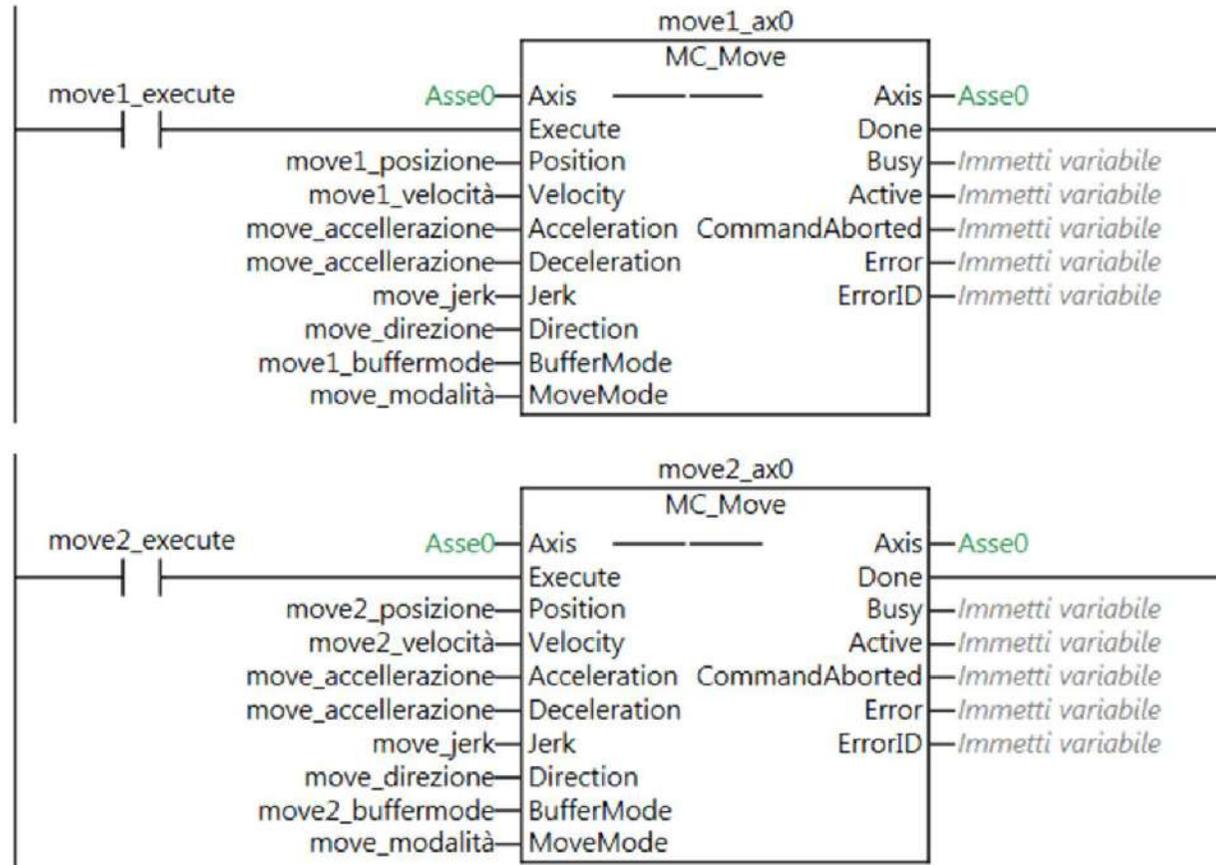
Parte 3: Movimentazione discreta

- Utilizzare 2 Function Block MC_Move per comandare due movimenti relativi
- L'esecuzione del movimento dovrà essere comandata prima che la movimentazione precedente abbia termine (verrà utilizzato il buffer di memoria integrato)

•Specifiche dei movimenti:

- Distanza 1: 180°
- Velocità 1: 270°/s
- Buffer Mode 1: `_mcBuffered`
- Accelerazione e decelerazione: 1000°/s²
- Jerk: 1000000°/s³
- Direzione di movimento: positiva
- Distanza 2: 180°
- Velocità 2: 135°/s
- Buffer Mode 2: `_mcBuffered`

Parte 3: Movimentazione discreta

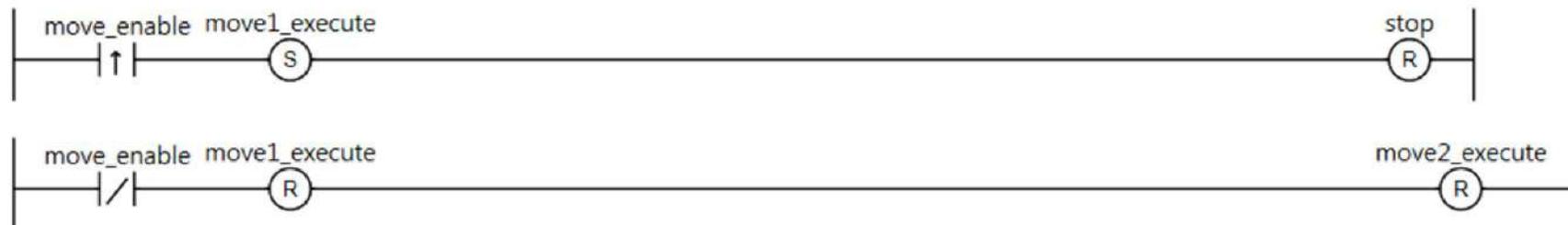


Parte 3: Movimentazione discreta

- Quando un movimento viene elaborato dal modulo MC (Active = TRUE), viene caricato nel buffer il movimento successivo.

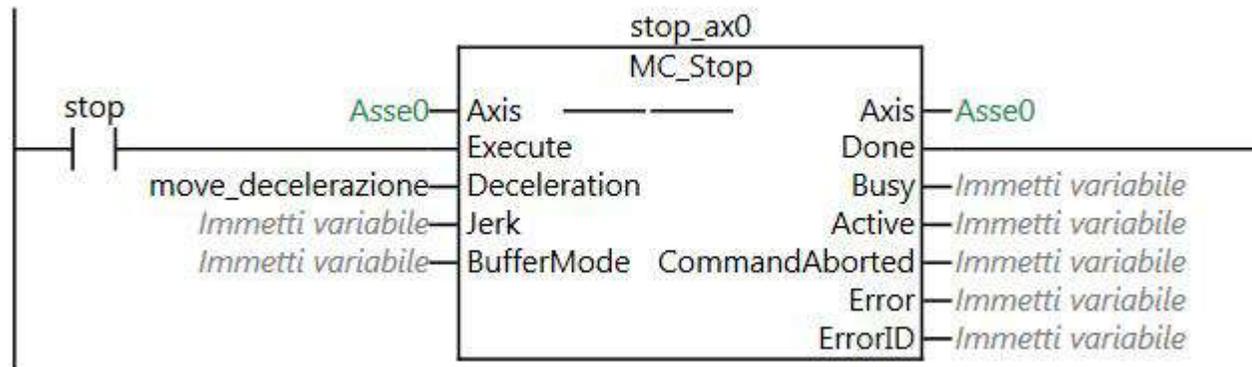


- La prima movimentazione viene attivata direttamente dal contatto «move_enable», che è anche in grado di fermare il processo.



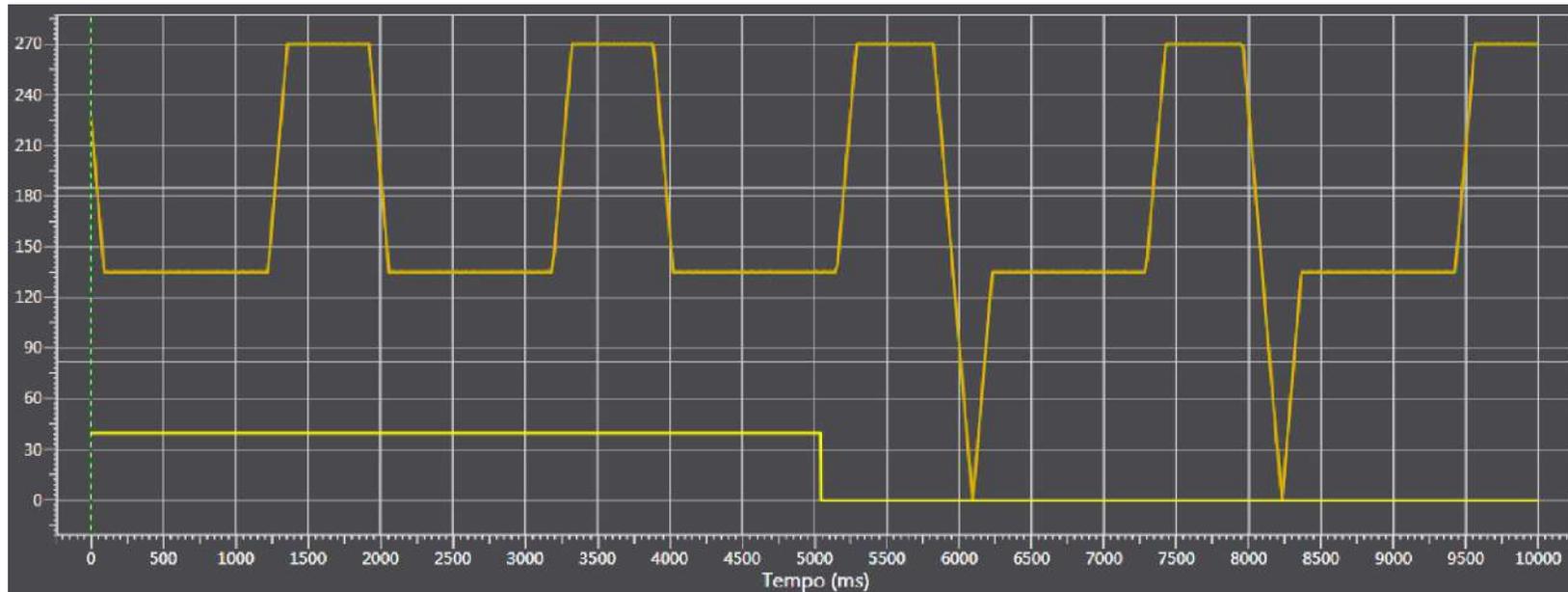
Parte 3: Interruzione del movimento

- Utilizzare il Function Block MC_Stop per fermare bruscamente il movimento



Parte 3: Blending Mode

- Variare il valore degli ingressi BufferMode e verificare come cambia il movimento dell'asse
- Aiutarsi con il Data Trace per visualizzare un grafico delle velocità





Esercizio #1

SYSTMAC
always in control

OMRON

Esercizio #1: modalità Blending

- Configurazione rete EtherCAT
- Configurazione degli assi
- Creazione di un programma che:
 - Abiliti entrambi gli assi
 - Esegua l'homing su entrambi gli assi
 - Esegua una coppia di movimenti in contemporanea su entrambi gli assi
 - Sull'asse MC_Axis001, dovrà essere attivata una delle modalità di Blending su entrambi i movimenti
 - Sull'asse MC_Axis000, dovrà essere attivata la modalità Buffering su entrambi i movimenti
- Effettuare il Data Trace delle velocità dei due assi

Esercizio #1: Configurazione dell'asse 0

- **Tipo asse:** Asse servoazionamento
- **Dispositivo d'ingresso:** Nodo 1
- **Contatore impulsi di comando:** 1048576 impulsi/rev
- **Distanza corsa:** 160.2212 mm/rev
- **Velocità massima:** 5000 mm/s
- **Velocità jog massima:** 3000 mm/s
- **Homing:** Ingresso prossimità approccio diretto OFF (su ingresso fase Z)
- **Velocità alta homing:** 1000 mm/s
- **Velocità bassa homing:** 50 mm/s

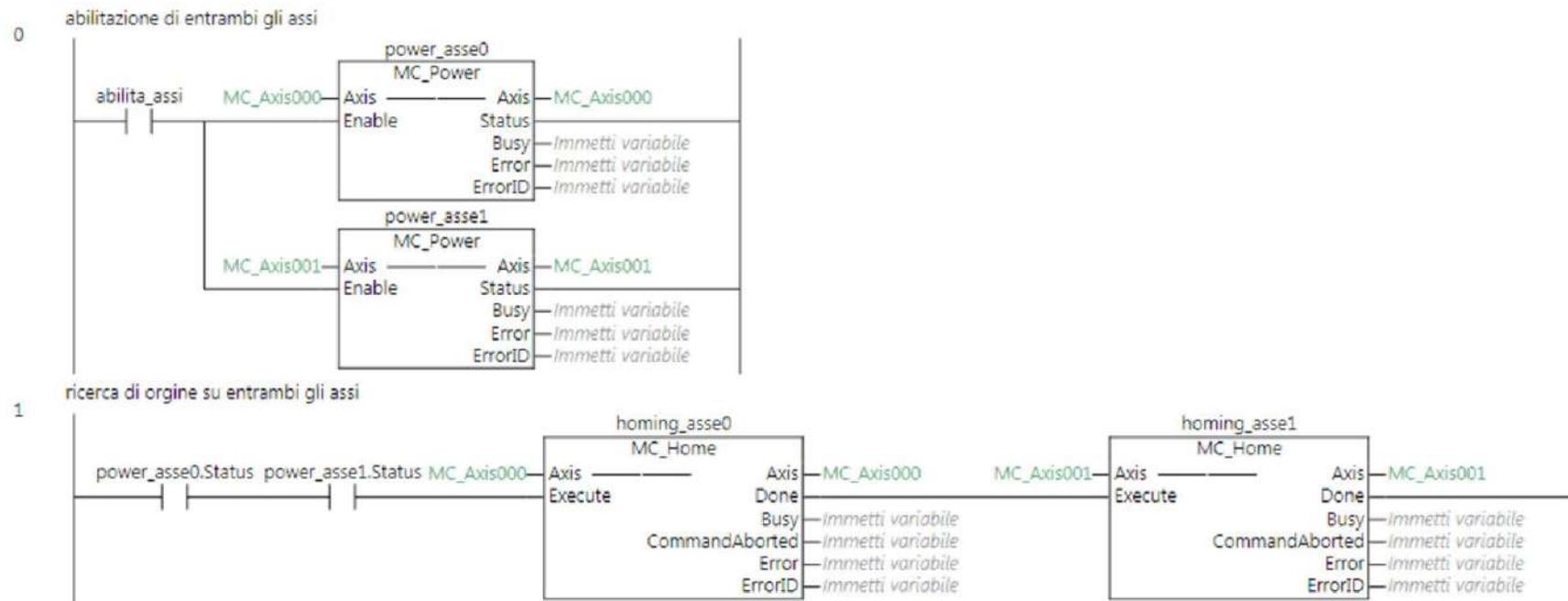
Esercizio #1: Configurazione dell'asse 1

- **Tipo asse:** Asse servoazionamento
- **Dispositivo d'ingresso:** Nodo 2
- **Contatore impulsi di comando:** 1048576 impulsi/rev
- **Distanza corsa:** 103.6725 mm/rev
- **Velocità massima:** 5000 mm/s
- **Velocità jog massima:** 3000 mm/s
- **Homing:** Distanza maschera ingresso (su ingresso esterno)
- **Velocità alta homing:** 500 mm/s
- **Velocità bassa homing:** 50 mm/s
- **Distanza maschera ingresso:** 10 mm

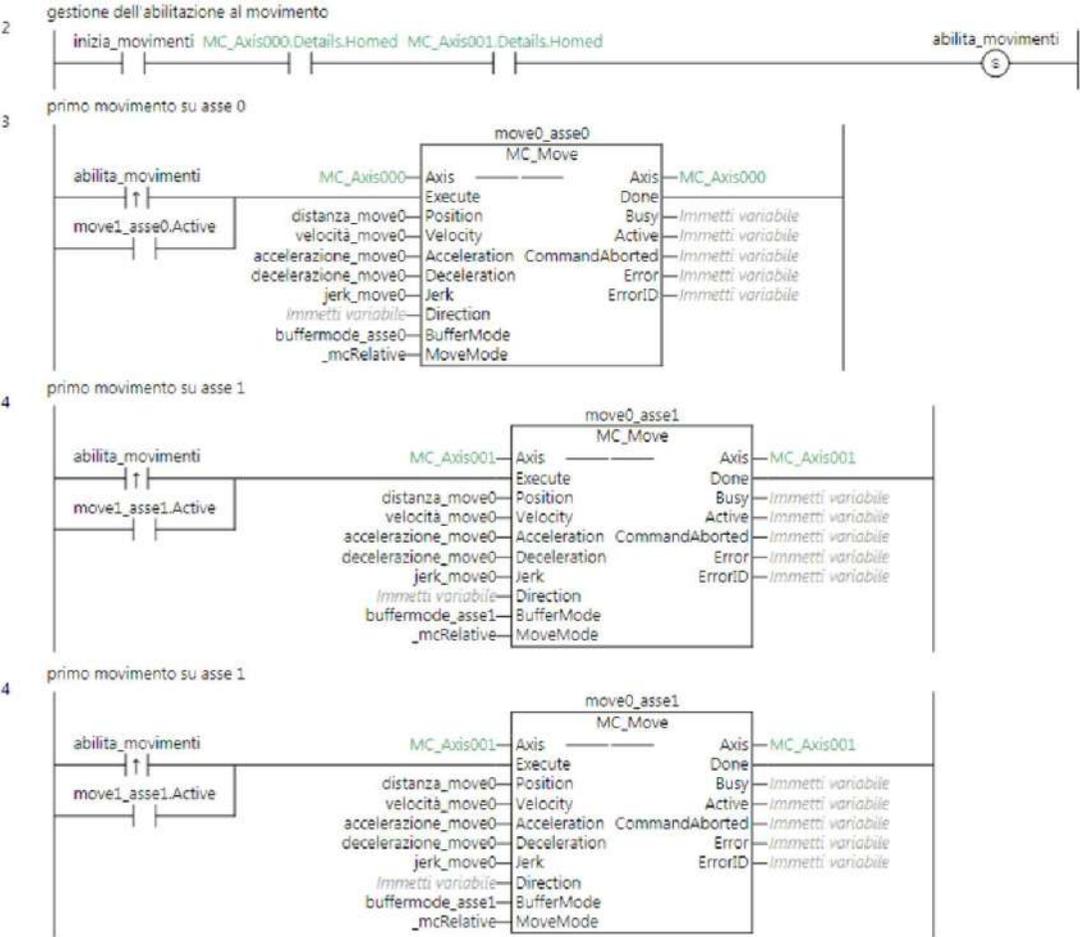
Esercizio #1: specifiche movimenti

- Primo movimento:
 - Posizionamento relativo di 200 mm
 - Velocità: 500 mm/s
 - Accelerazione e decelerazione: 5000 mm/s²
 - Limitazione di jerk disabilitata
- Secondo movimento:
 - Posizionamento relativo di 200 mm
 - Velocità: 100 mm/s
 - Accelerazione e decelerazione: 2000 mm/s²
 - Limitazione di jerk disabilitata

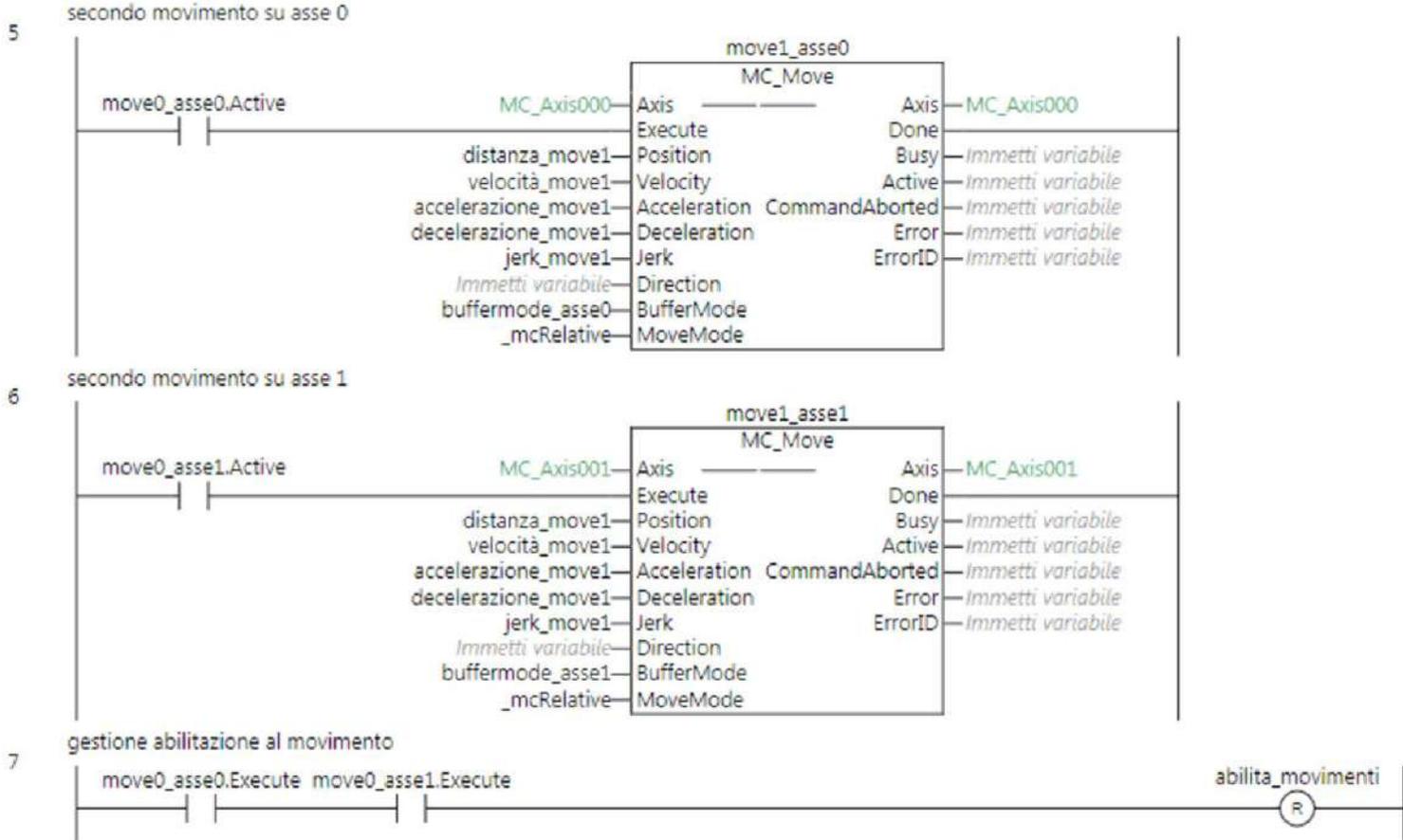
Esercizio #1



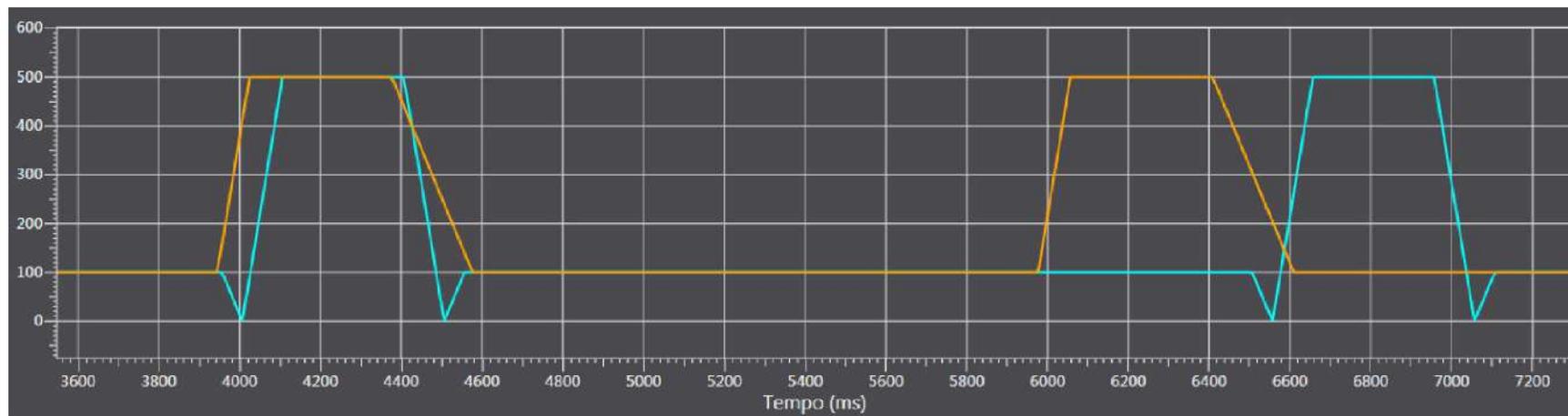
Esercizio #1



Esercizio #1

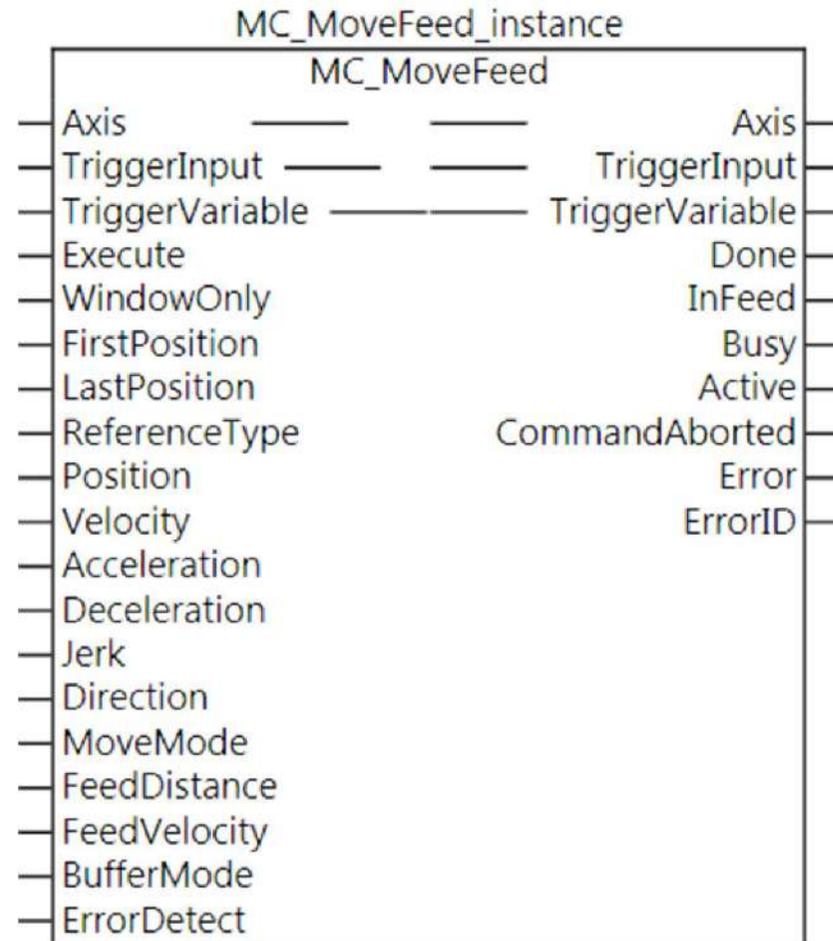


Esercizio #1



MC_MoveFeed

- L'istruzione MC_MoveFeed opera un posizionamento all'intervento di un trigger esterno.
- Se, durante il trigger, è in corso una movimentazione (assoluta, relativa o un comando di velocità), viene annullata in favore del posizionamento della MoveFeed.
- Nelle variabili **Position**, **Velocity**, **Acceleration**, **Deceleration**, **Jerk**, **Direction** e **MoveMode** occorre inserire il movimento primario che verrà interrotto all'arrivo del trigger.



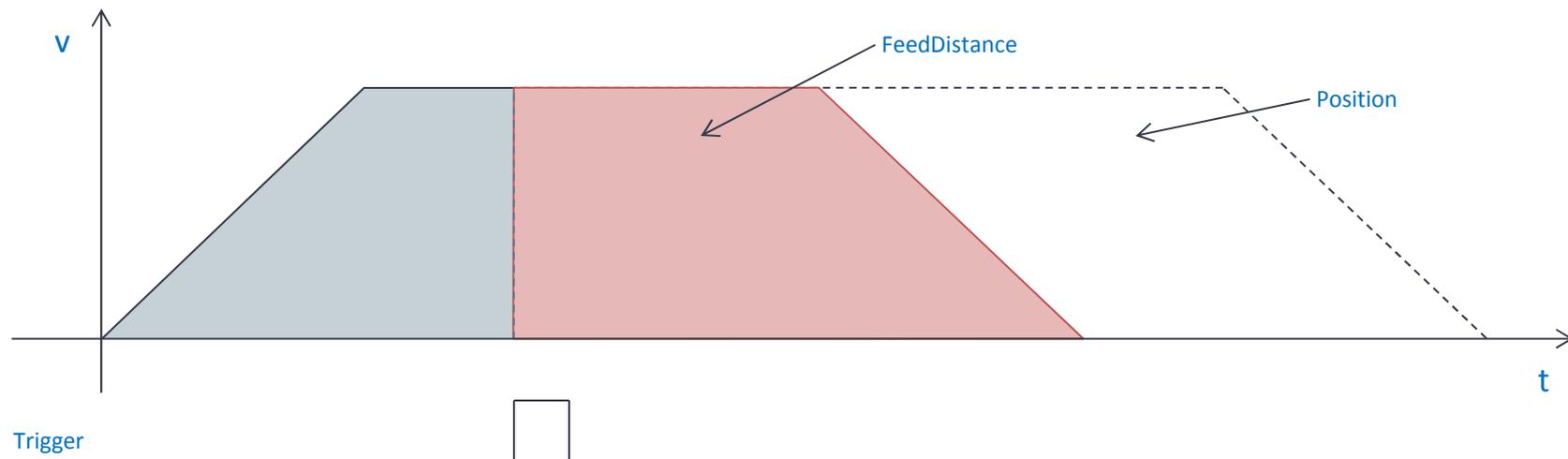
MC_MoveFeed

Mode	InputDrive	LatchID	Descrizione
_mcController	X	X	Variabile TriggerVariable
_mcDrive	_mcEncoderMark	X	Fase Z dell'encoder motore
_mcDrive	_mcEXT	_mcLatch1	Ingresso Latch1 dello slave
_mcDrive	_mcEXT	_mcLatch2	Ingresso Latch2 dello slave

- La definizione del trigger è effettuata dalla variabile struct **TriggerInput**, come mostra la tabella.
- Il movimento che viene avviato in seguito all'intervento del trigger è definito dalle variabili **FeedDistance** e **FeedVelocity**.

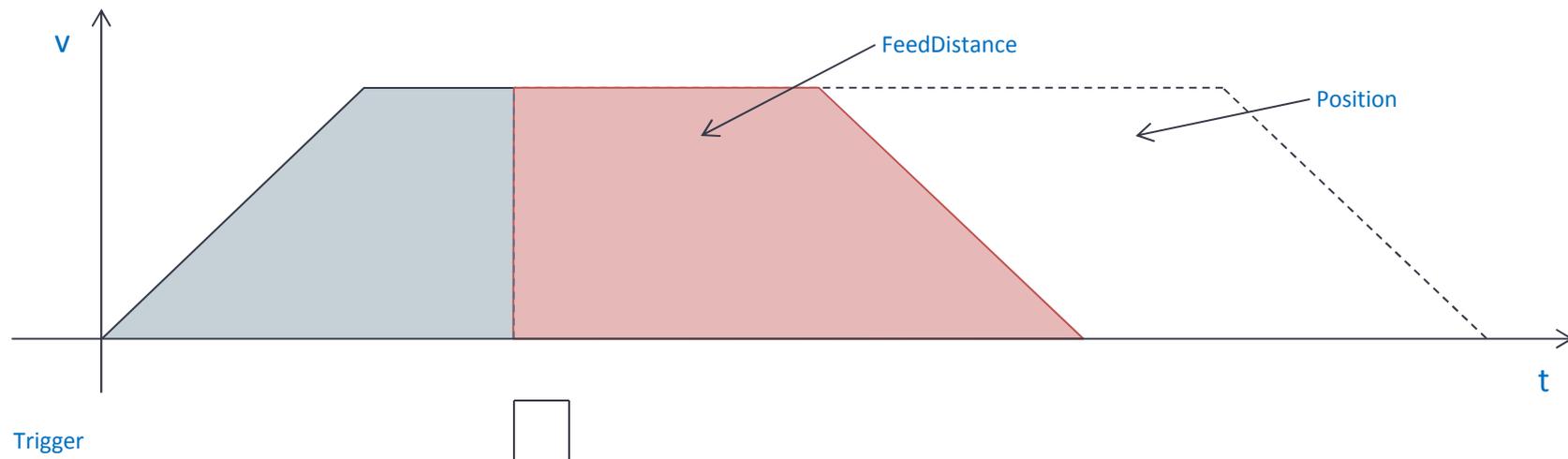
MC_MoveFeed

- All'intervento del trigger la prima movimentazione viene fermata, in favore della seconda: dal trigger alla fermata verrà percorso lo spazio FeedDistance.
- Se non si dovesse verificare un trigger, la movimentazione si fermerebbe quando l'asse raggiunge Position



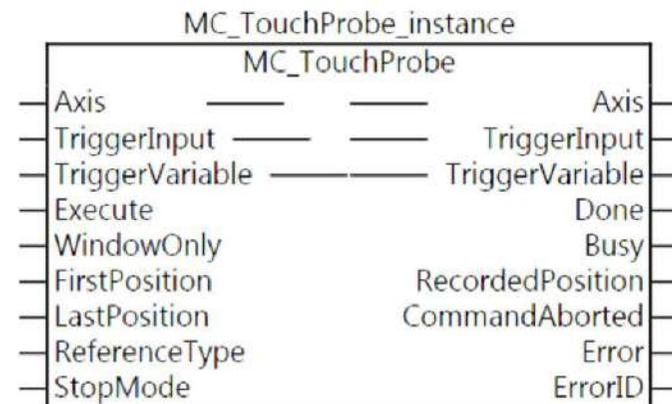
MC_MoveFeed

- Mediante le variabili **WindowOnly**, **FirstPosition** and **LastPosition**, è possibile indicare una finestra di intervento del trigger.
- Se il trigger dovesse avvenire al di fuori di questa finestra, la movimentazione di Feed non avrebbe luogo.



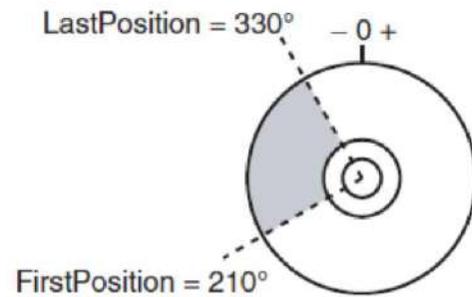
MC_TouchProbe

- L'istruzione **MC_TouchProbe** è in grado di effettuare una registrazione veloce e precisa della posizione raggiunta dall'asse all'intervento di un trigger esterno.
- In modo analogo alla funzione MC_MoveFeed, è possibile utilizzare 3 tipi di trigger:
 - Ingressi veloci del servodrive (Latch1 e Latch2)
 - Fase Z del feedback (encoder motore o asse encoder)
 - Variabile interna (TriggerVariable)
- È inoltre possibile impostare una finestra di acquisizione, tramite le variabili WindowOnly, FirstPosition e LastPosition, ed un metodo di arresto qualora la misura venga acquisita.

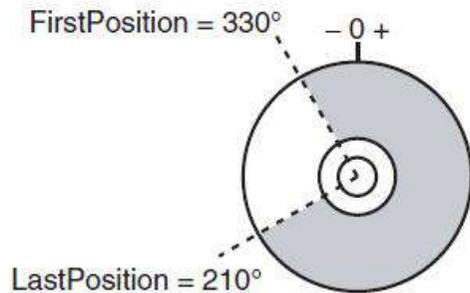


MC_TouchProbe

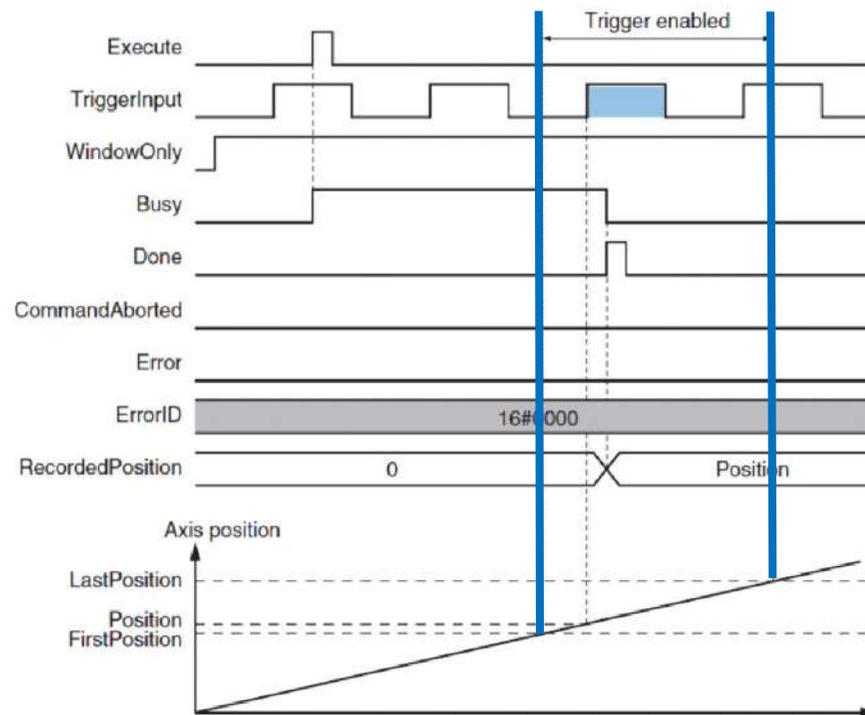
- Quando la finestra di acquisizione è abilitata (WindowOnly = TRUE), la posizione è catturata solo se il trigger è presente nell'area definita:



FirstPosition < LastPosition



FirstPosition > LastPosition

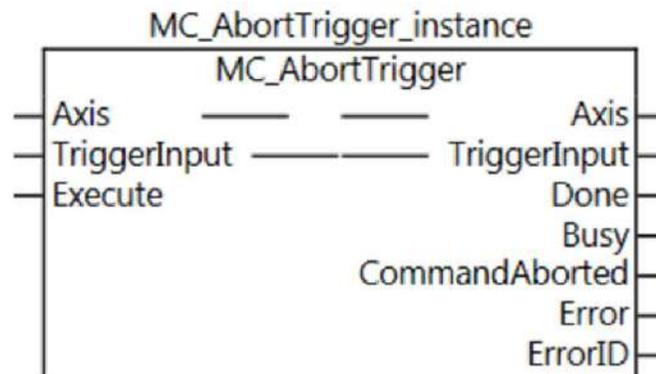


MC_TouchProbe

- La variabile StopMode indica il comportamento dell'asse dopo l'acquisizione:
 - Specificando **_mcNonStop**: una volta effettuata l'acquisizione l'asse proseguirà il suo movimento
 - Specificando **_mcImmediateStop**: l'asse si fermerà nella posizione acquisita, mediante Stop Immediato.
- Questa funzionalità è permessa solamente nella modalità CSP del drive: se dovesse essere eseguita nelle modalità CSV o CST, viene mostrato un errore.

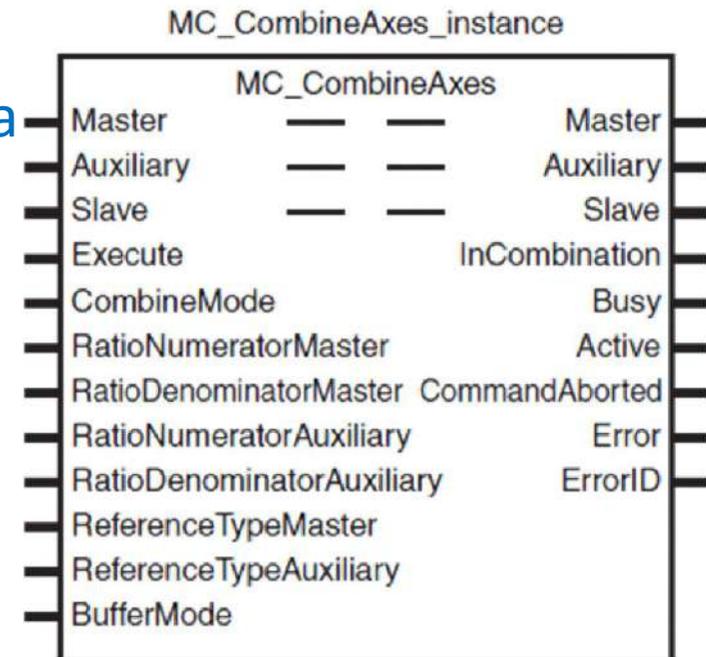
MC_AbortTrigger

- L'istruzione **MC_AbortTrigger** è necessaria per annullare la richiesta di trigger, attivata dalle istruzioni MC_MoveFeed e MC_TouchProbe:

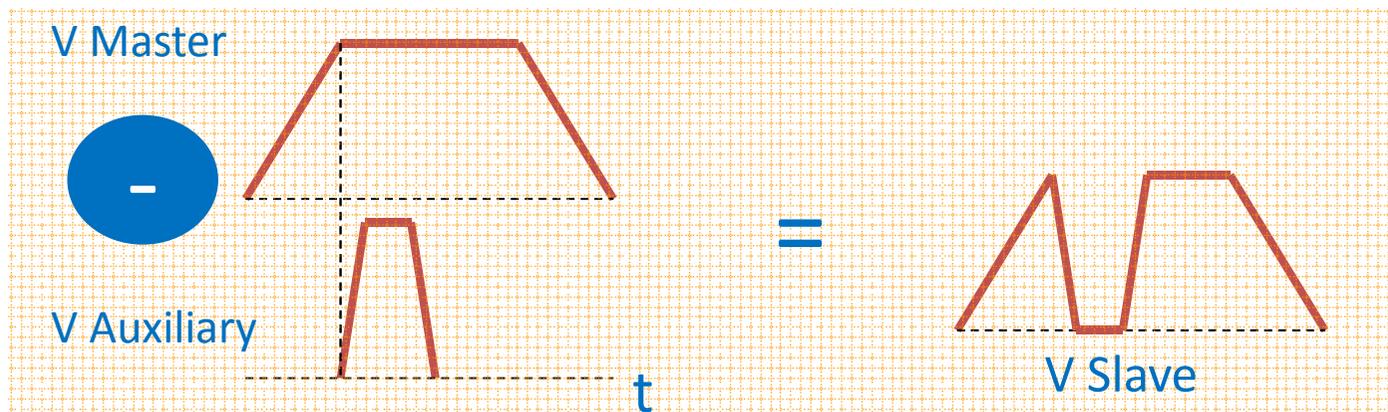
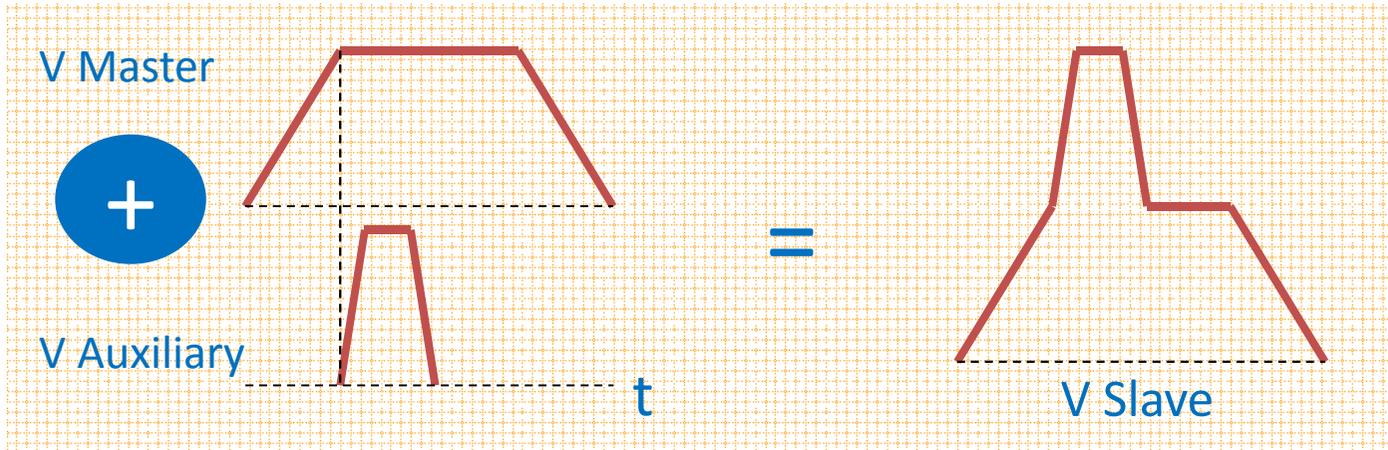


MC_CombineAxes

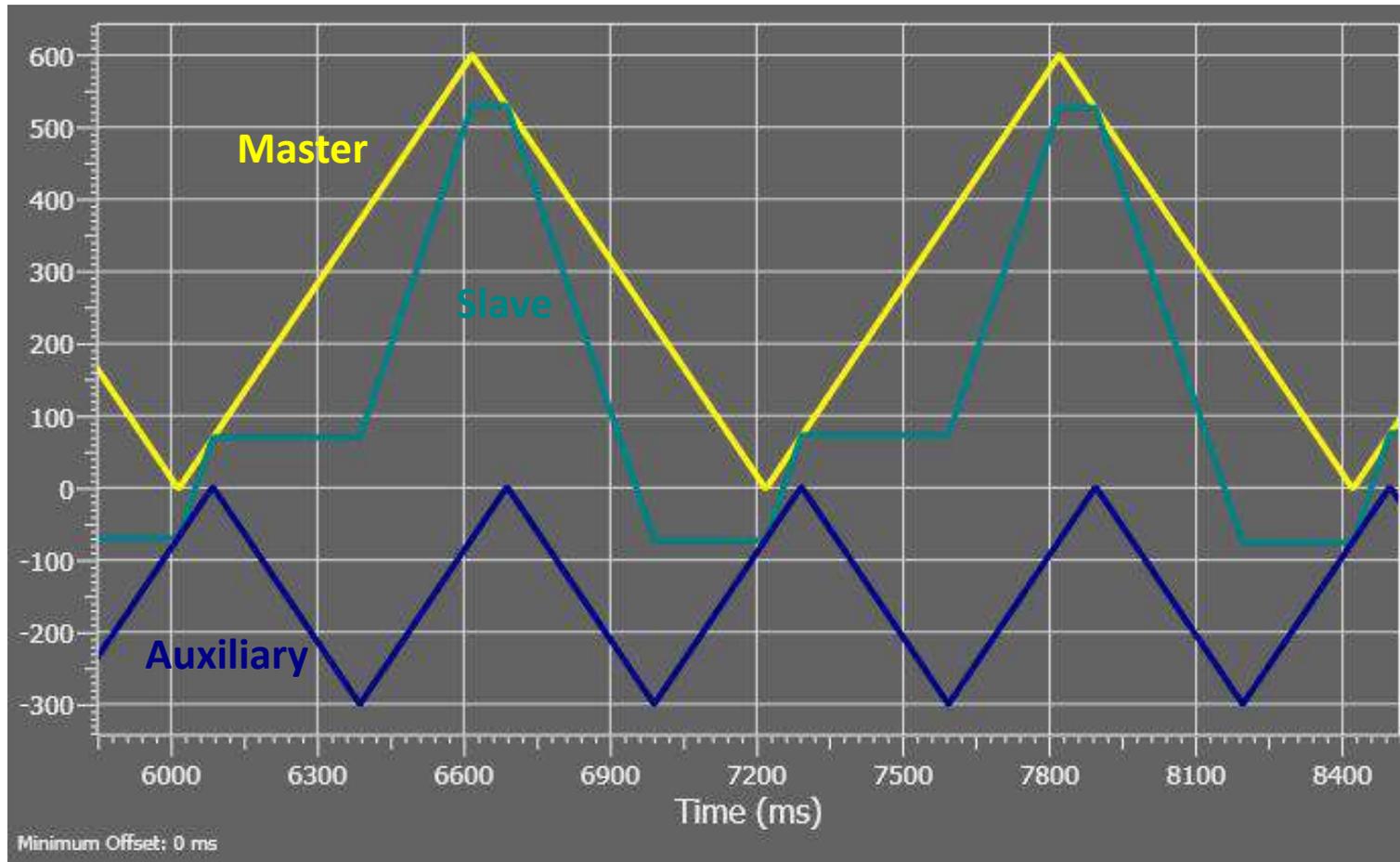
- L'istruzione **MC_CombineAxes** genera il profilo di posizione di un asse Slave, come **somma** o **sottrazione** tra profili degli assi Master ed Ausiliario.
- La variabile **CombineMode** seleziona l'operazione da svolgere (somma o sottrazione)
- **RatioNumeratorMaster**, ..., **RatioDenominatorAuxiliary**, permettono la scalatura dei profili Master edAuxiliary
- **ReferenceType** indica la natura (comando o feedback) dei profili da combinare



MC_CombineAxes



MC_CombineAxes





Esercizio #1

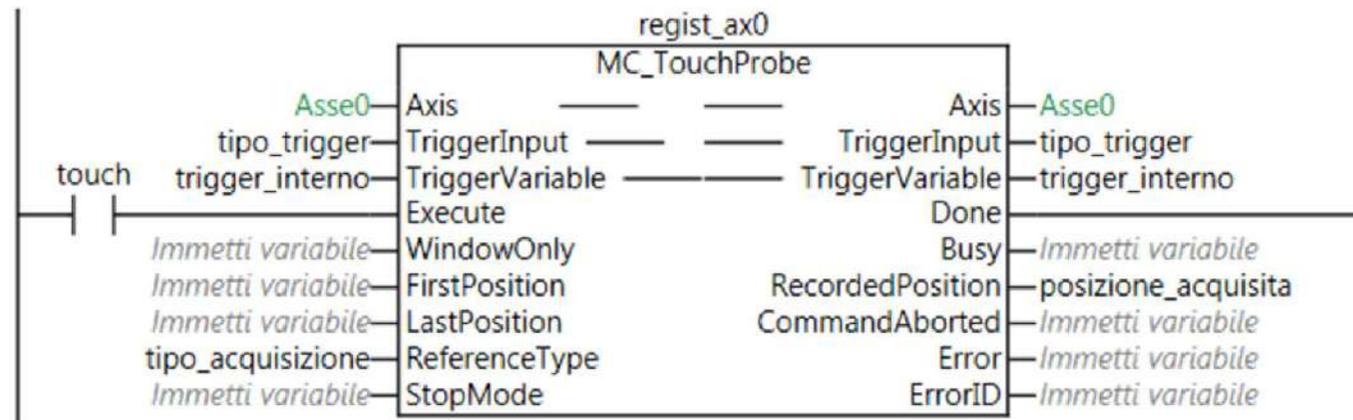
(Parte 4)

SYSTMAC
always in control

OMRON

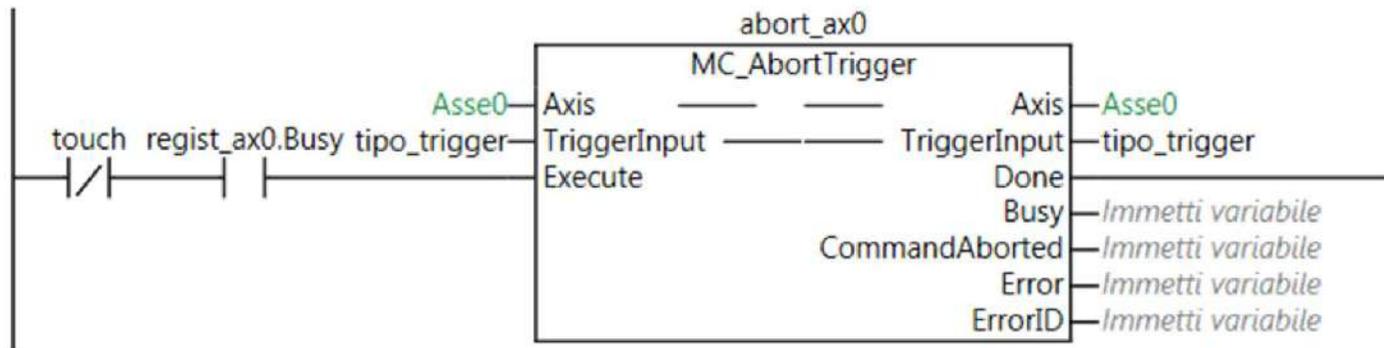
Parte 4: Registrazione rapida della posizione

- Utilizzare il Function Block MC_TouchProbe per effettuare una registrazione veloce della posizione del feedback, tramite un trigger interno (TriggerVariable).
- Il risultato dell'acquisizione verrà indicato nell'uscita **RecordedPosition**.



Parte 4: Interruzione della registrazione rapida

- Utilizzare il Function Block MC_AbortTrigger interrompere la chiamata della registrazione rapida.





Domande?

E OY!



SYSTMAC
always in control

OMRON



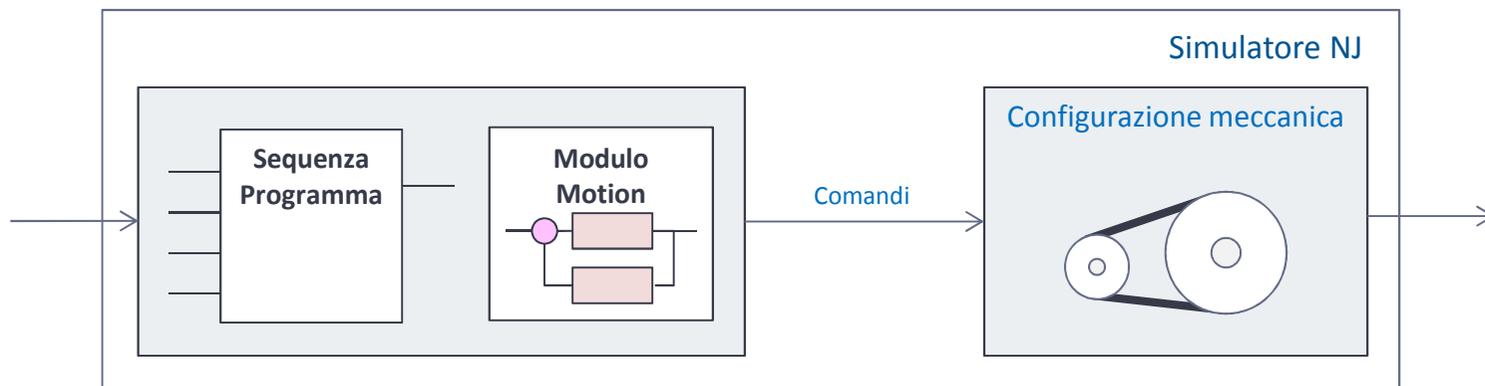
Simulatore

SYSTMAC
always in control

OMRON

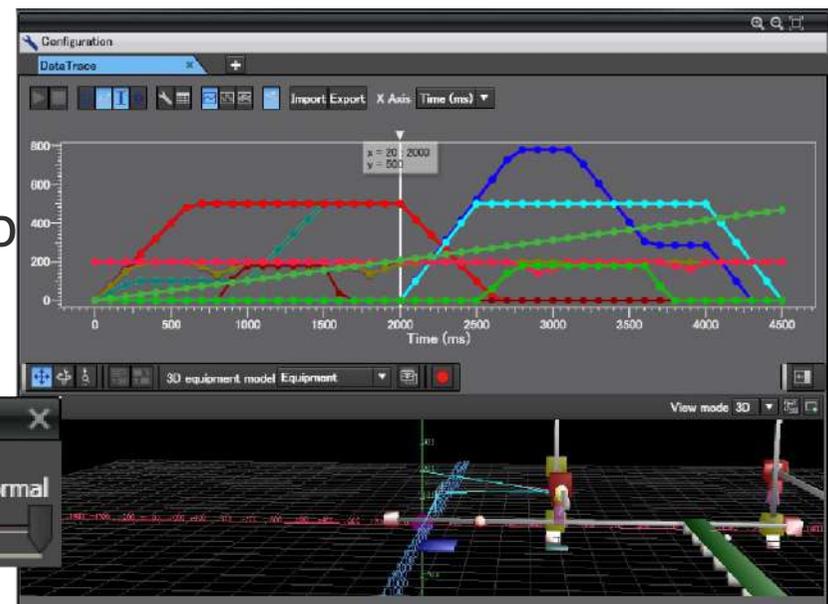
Funzionalità del simulatore integrato

- Il simulatore integrato **in** Sysmac Studio permette di effettuare una simulazione completa del programma, comprensiva di logica e motion
- Indicando la configurazione meccanica della macchina, è possibile effettuare simulazioni della traiettoria 2D e 3D, inerente a comandi di interpolazione su gruppi assi



Funzionalità del simulatore integrato

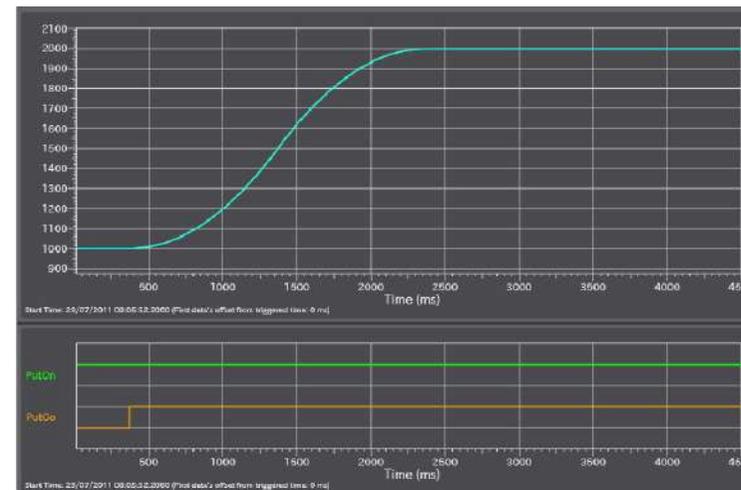
- Il simulatore offre la possibilità di verificare:
- Sincronizzazione simulata di Motion e Sequenza logica
- Comparazione tra valori di comando e valori di feedback (stimati)
- Debugging veloce che può fermare l'operazione velocemente in una scena specifica
- Stima del tempo di esecuzione
- Simulazione del servoazionamento (POT, NOT, EXT1, ...)



Data trace in simulazione

- Durante la simulazione, è possibile effettuare Data Trace delle variabili di comando e di feedback degli assi e delle variabili utilizzate nella sequenza logica
- Tutte le variabili sono sincronizzate temporalmente

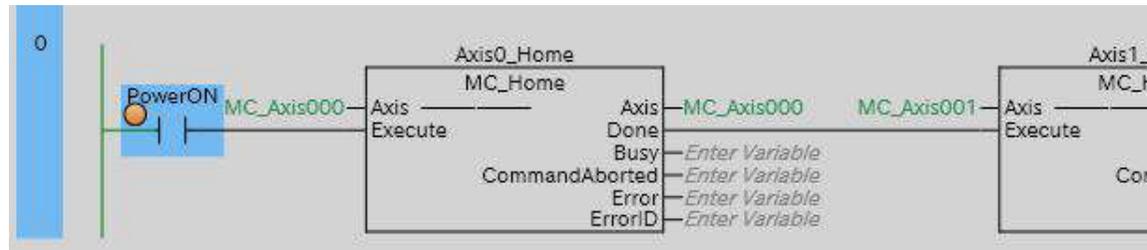
Visible		Name	X Offset	Y Offset
<input checked="" type="checkbox"/>		MC_Axis000.Cmd.Pos	0	0
<input checked="" type="checkbox"/>		MC_Axis000.Act.pos	0	0
<input checked="" type="checkbox"/>		Program0.PutOn	0	
<input checked="" type="checkbox"/>		Program0.PutGo	0	



Funzioni di debug

- **Breakpoints** ed **esecuzione a step** possono essere **utilizzati** per effettuare il debug del programma

```
2
3 varA:=13;
4 varB:=69;
5
6 IF varA < DINT#1 THEN
7   varA:=DINT#100;
8 END_IF;
9
10 IF varA > DINT#12 THEN
11   varA:= DINT#200;
12 END_IF;
13
14 FOR varB:= 0 TO 2 BY 1 DO
15   varA:=varB;
16 END_FOR;
17
```



- Anche in simulazione è possibile utilizzare la **finestra Watch Tab** per modificare e monitorare le variabili del programma

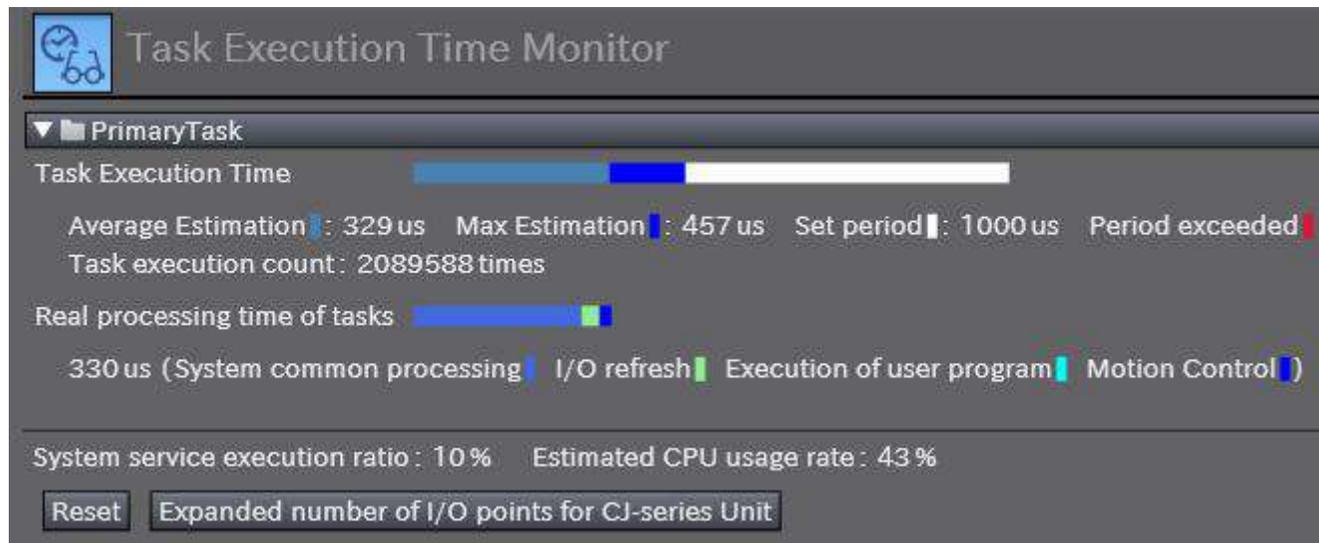
Name	Online value	Modify	Data type	Address
GroupEnable	True	TRUE FALSE	BOOL	

Funzioni di debug

- Speciali task eseguiti solo in debug possono risultare utili per la simulazione di fenomeni esterni al controllore:
 - Timing ingressi digitali ed analogici
 - Ingressi dei servoazionamenti (POT, NOT, Origine, ...)
 - Condizioni di allarme o warning da parte del servoazionamento

Simulazione del tempo di esecuzione

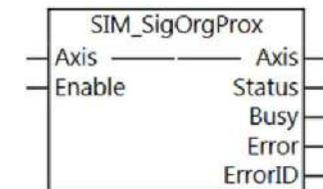
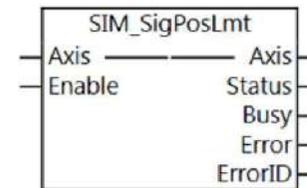
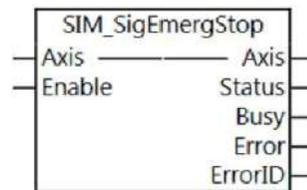
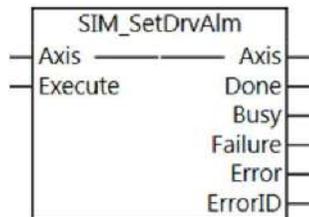
- Il simulatore è in grado di effettuare una stima del tempo di esecuzione di ogni singolo task:



Simulazione del tempo di esecuzione

- Il tempo di esecuzione stimato include gli effetti delle prestazioni del computer sul quale si sta eseguendo la simulazione.
- Variazioni del tempo di esecuzione, dovute a salti condizionati, possono essere controllate: il simulatore calcolerà tempi minimi, medi e massimi.

- Per simulare questi comportamenti, sono **presenti** FB dedicati alla simulazione.



- Il task di debug non viene scaricato nel controllore una volta che l'NJ viene sincronizzato

Data Trace

The screenshot shows the 'Registrazione dati' (Data Recording) window. The interface includes a toolbar with icons for manual trigger, start/stop, visualization settings, and acquisition display. Below the toolbar, there are configuration options for recording type (Singolo), sampling interval (Tutti i periodi di task), and primary task. A table lists visible variables for tracing, including 'Asse_x.Act.Pos', 'Asse_x.Cmd.Pos', 'Asse_y.Act.Pos', and 'Asse_y.Cmd.Pos'. At the bottom, a graph displays the recorded data over time, with a callout box indicating how to add variables to be traced.

Trigger manuale

Start/Stop

Impostazioni di visualizzazione

Mostra acquisizione

Selezione tipo di trace

Acquisizione in tempo reale

Aggiungi variabili da tracciare

Visibile	Nome	Offset X	Offset Y	Min. asse Y	Max asse Y	Minimo	Massimo	Med
<input type="checkbox"/>	Asse_x.Act.Pos	0	0	0	0	0	1100	541.18655624824
<input checked="" type="checkbox"/>	Asse_x.Cmd.Pos	0	0	0	0	0	1100	541.1865833172045
<input type="checkbox"/>	Asse_y.Act.Pos	0	0	0	0	-50	600	274.5535479068756
<input checked="" type="checkbox"/>	Asse_y.Cmd.Pos	0	0	0	0	-50	600	274.5535902045729

Configurazione meccanica

Aggiungi/Cancela modelli macchina 3D

Tipo di modello macchina 3D

Impostazioni modello macchina 3D

Nome	Tipo dati	Valore *	Converti valori	modulo
Trasportatore:Variabile corrispondente	_sAXIS_REF	Asse_x		
Trasportatore:Lunghezza	LREAL	1000		

Posizione attuale

X: 870
Y: -20
Z: -140

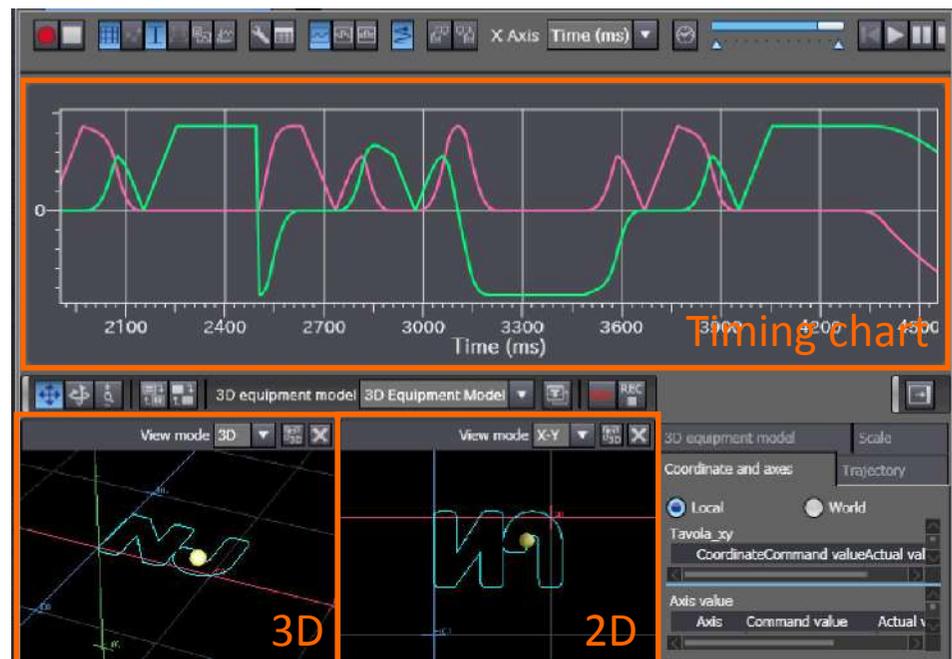
Orientamento

Rx(deg): 0
Ry(deg): 0
Rz(deg): 270

Posizione ed orientamento del modello

Visualizzazione 2D/3D

- Il Data Trace è in grado di visualizzare il risultato della simulazione da vari punti di vista, aggiungendo viste 2D e 3D.
- La simulazione permette la sincronizzazione tra grafici temporali e visualizzazioni 2D e 3D.



Comparazione comandi e feedback

- I valori di comando e di feedback possono essere comparati all'interno della vista 2D/3D

Visible		Name	X Offset	Y Offset	Y Axis Min.	Y Axis Max.
<input type="checkbox"/>		Asse_x.Cmd.Pos	0	0	0	0
<input type="checkbox"/>		Asse_x.Act.Pos	0	0	0	0
<input type="checkbox"/>		Asse_y.Cmd.Pos	0	0	0	0
<input type="checkbox"/>		Asse_y.Act.Pos	0	0	0	0
<input checked="" type="checkbox"/>		Asse_x.cmd.vel	0	0	0	0
<input checked="" type="checkbox"/>		Asse_y.cmd.vel	0	0	0	0

- Quando si assegna una variabile **del tipo `_sAXIS_REF`** ad un asse del modello 3D, le grandezze relative al comando di posizione ed il feedback di posizione vengono automaticamente inserite alle variabili da tracciare



- Nelle impostazioni della traiettoria è possibile visualizzare i valori del comando e/o i valori del feedback

Finestra di monitoraggio variabili

- Tramite la finestra di monitoraggio variabili, è possibile:
 - Cambiare i valori delle variabili in ingresso
 - Visualizzare i valori delle variabili simulate in uscita

The screenshot shows the 'Watch Window (Control)' interface with the following table:

Name	Online value	Modify	Data type	AT	Data format
Program0.linear_busy	True	TRUE FALSE	BOOL		Boolean
Program0.power	True	TRUE FALSE	BOOL		Boolean
Program0.k	6		INT		Decimal
Program0.circular_busy	True	TRUE FALSE	BOOL		Boolean
<i>Input Name...</i>					

Callouts from the image:

- Nome della variabile:** Points to the 'Name' column.
- Tipo dati della variabile:** Points to the 'Data type' column.
- Formato di visualizzazione:** Points to the 'Data format' dropdown menu.
- Valore simulato attuale:** Points to the 'Online value' column.
- Area di variazione delle variabili:** Points to the 'Modify' column.

Differenze tra simulazione e esecuzione normale

- L'appendice del manuale W504 contiene tutte le differenze.
- Di seguito le differenze più significative
- Scheduling: non sono rispettate le priorità
- Monitor: viene mostrato il tempo stimato di esecuzione del programma
- Dati: I valori delle variabili di programma sono resettati quando si esce dalla simulazione
- Dati: I valori retentivi non sono mantenuti
- Debugger: può essere usato solo in simulazione
- Motion: gli assi reali sono simulati
- Motion: la quota attuale è presa dalla quota di comando
- Network: la comunicazione di rete non è eseguita in simulazione



Esercizio #2

(Parte 1)

SYSTMAC
always in control

OMRON

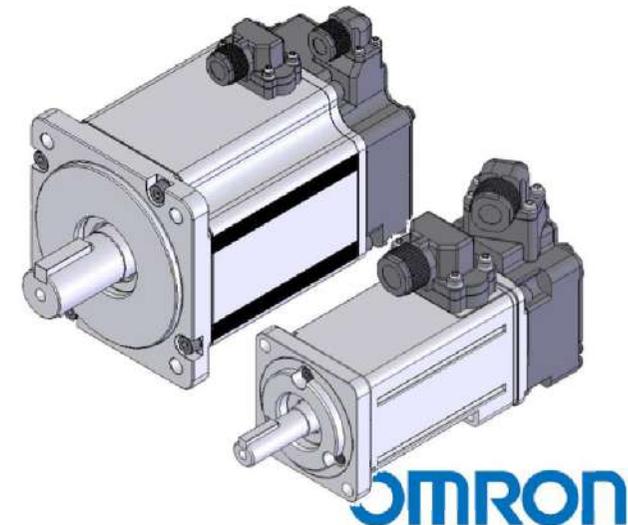
Parte 1: Configurazione della rete EtherCAT

- Comporre una rete EtherCAT costituita da un tre nodi slave
- Tutti gli slave dovranno essere servoazionamenti G5 da 100W 200V, con versione firmware 2.1 (codice: R88D-KN01H-ECT Rev:2.1)



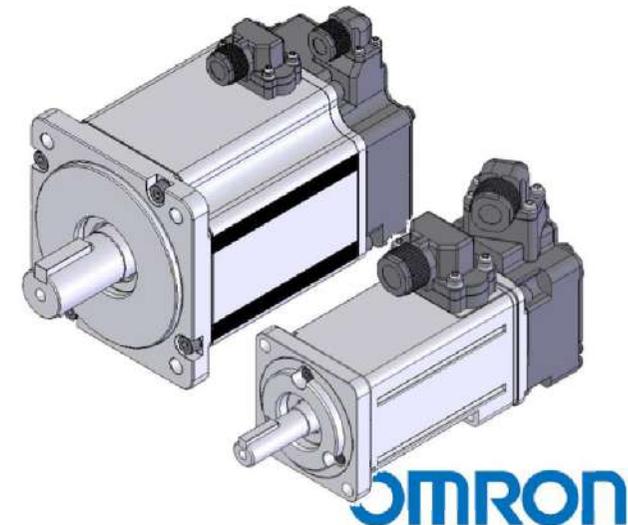
Parte 1: Configurazione degli assi

- **Nome asse:** Nastro
- **Tipo asse:** Asse servoazionamento (assegnato al Nodo 1)
- **Contatore impulsi di comando:** 1048576 impulsi/rev
- **Distanza corsa:** 10 mm/rev
- **Velocità massima:** 3000 rpm (=500 mm/s)
- **Velocità jog massima:** 2100 rpm (=350 mm/s)
- **Homing:** Posizione zero preimpostata
- **Modalità di conteggio:** lineare
- **Tipo di encoder:** incrementale



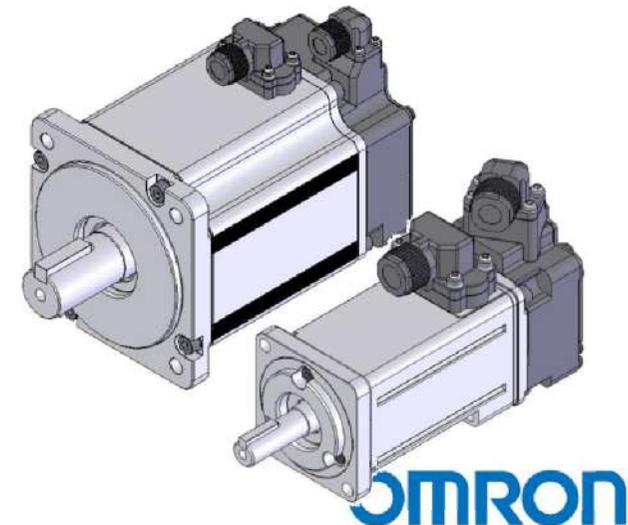
Parte 1: Configurazione degli assi

- **Nome asse:** Lama_oriz
- **Tipo asse:** Asse servoazionamento (assegnato al Nodo 2)
- **Contatore impulsi di comando:** 1048576 impulsi/rev
- **Distanza corsa:** 100 mm/rev
- **Velocità massima:** 3000 rpm (=5000 mm/s)
- **Velocità jog massima:** 900 rpm (=1500 mm/s)
- **Homing:** Solo ingressi limite
- **Velocità homing:** 500 mm/s
- **Velocità di avvicinamento:** 50 mm/s
- **Modalità di conteggio:** lineare
- **Tipo di encoder:** incrementale



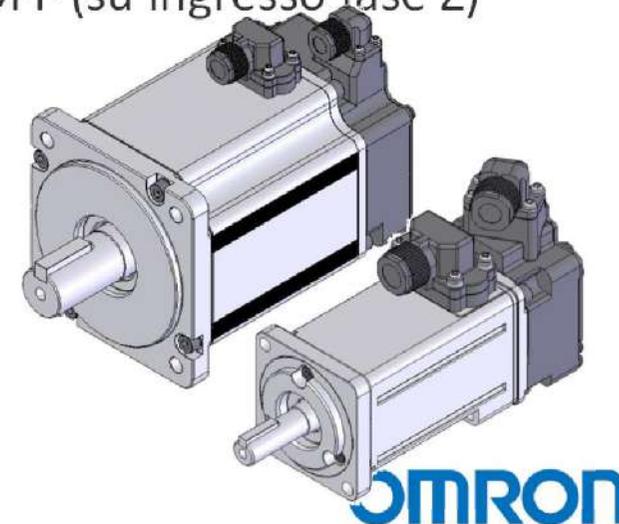
Parte 1: Configurazione degli assi

- **Nome asse:** Lama_vert
- **Tipo asse:** Asse servoazionamento (assegnato al Nodo 3)
- **Contatore impulsi di comando:** 1048576 impulsi/rev
- **Distanza corsa:** 100 mm/rev
- **Velocità massima:** 3000 rpm (=5000 mm/s)
- **Velocità jog massima:** 900 rpm (=1500 mm/s)
- **Homing:** Posizione zero preimpostata
- **Modalità di conteggio:** lineare
- **Tipo di encoder:** incrementale



Parte 1: Configurazione degli assi

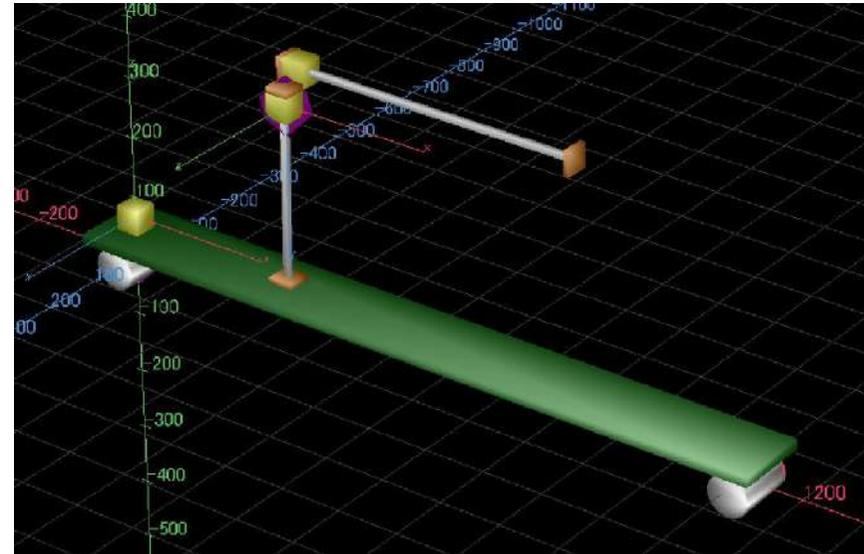
- **Nome asse:** Lama_vert
- **Tipo asse:** Asse servoazionamento (assegnato al Nodo 3)
- **Contatore impulsi di comando:** 1048576 impulsi/rev
- **Distanza corsa:** 100 mm/rev
- **Velocità massima:** 3000 rpm (=5000 mm/s)
- **Velocità jog massima:** 900 rpm (=1500 mm/s)
- **Homing:** Ingresso prossimità approccio diretto OFF (su ingresso fase Z)
- **Modalità di conteggio:** lineare
- **Tipo di encoder:** incrementale



Parte 1: Costruzione del modello da simulare

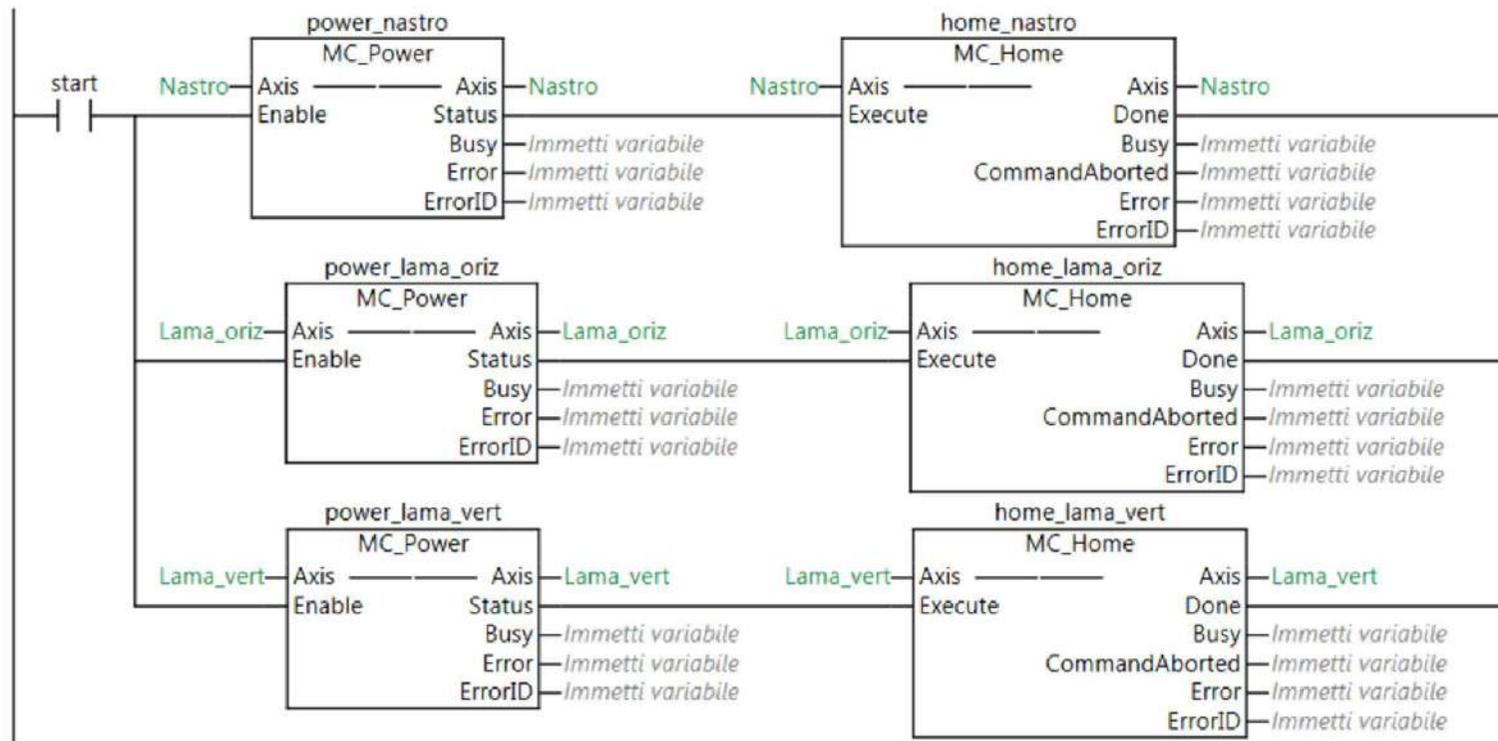
- **Tipo:** Trasportatore
- **Assegnazione asse:** Nastro
- **Lunghezza:** 1200 mm

- **Tipo:** Robot ortogonale (XY)
- **Assegnazione asse X:** Lama_oriz
- **Lunghezza asse X:** 500 mm
- **Assegnazione asse Y:** Lama_vert
- **Lunghezza asse Y:** 300 mm
- **Posizione:** X = 300 mm, Y = 0 mm, Z = 340 mm
- **Orientamento:** Rx = 90°, Ry = 0°, Rz = 0°



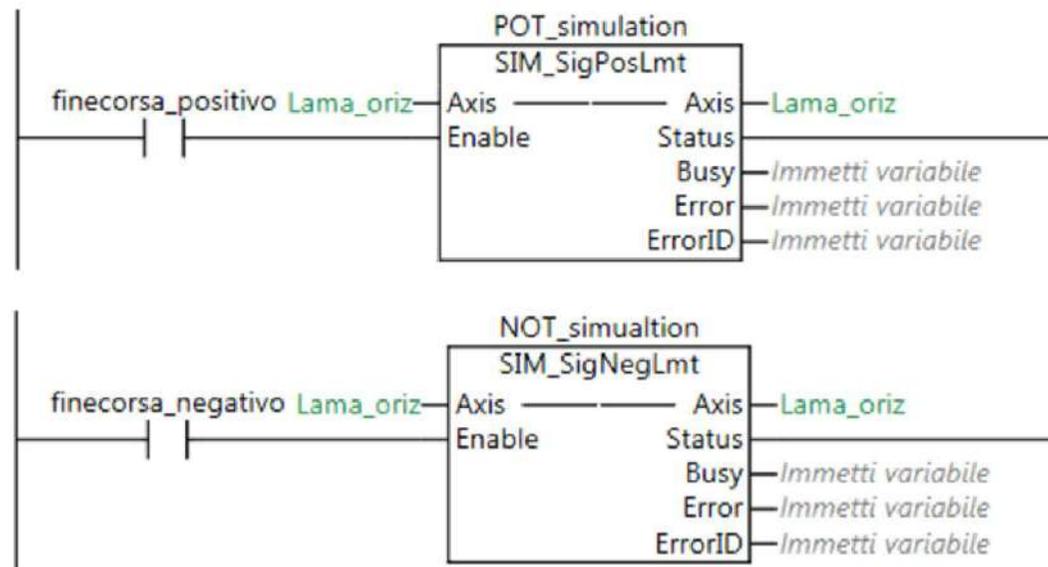
Parte 1: Messa in coppia e ricerca degli zeri

- Costruire un programma che metta in coppia tutti gli assi e che effettui la ricerca degli zeri.



Parte 1: Simulazione dei finecorsa

- Aggiungere ai POU un programma di debug, assegnato al task primario.
- Utilizzare le istruzioni **SIM_SigPosLmt** e **SIM_SigNegLmt** per simulare l'intervento dei finecorsa.





Domande?

E OY!

SYSTMAC
always in control

OMRON



SYSTMAC
always in control

OMRON