



## **Automazione industriale dispense del corso 18. Modellizzazione con SFC**

Luigi Piroddi  
[piroddi@elet.polimi.it](mailto:piroddi@elet.polimi.it)

## Introduzione

In generale, è bene cominciare a progettare le specifiche in modo *funzionale*, descrivendo (ad un livello di astrazione elevato) la combinazione di comandi e misure necessari per svolgere le operazioni del processo nell'ordine corretto.

Si descrive cioè in modo *formale* il comportamento desiderato, senza entrare in dettagli implementativi:

- ▶ comandi e misure sono specificati a parole
- ▶ non si entra nello specifico di come si fa ad ottenere una misura o di come si fa a generare fisicamente un comando

Il modello formale del comportamento desiderato ci consente di evidenziare:

- ▶ errori concettuali
- ▶ specifiche incomplete
- ▶ inconsistenze

Il modello formale del comportamento desiderato sarà parte dell'algoritmo di controllo finale.

Il modello funzionale va poi completato in uno schema più dettagliato, di livello *operazionale*, in cui alle azioni e alle misure espresse in modo descrittivo nel linguaggio naturale vanno sostituiti i costrutti formali che fanno riferimento ai segnali effettivamente a disposizione, tenendo conto della tecnologia utilizzata per l’attuazione del controllo e per il rilevamento delle misure.

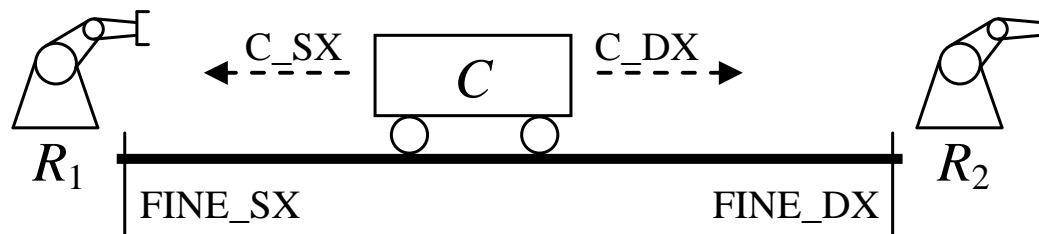
Procedimento tipico:

- ▶ definizione del modello funzionale, che rappresenta la logica di controllo principale del sistema
- ▶ eventuale espansione dei passi (approccio top-down)
- ▶ definizione del modello operativo
- ▶ eventuale espansione dei passi (approccio top-down)
- ▶ verifica tramite simulazione
- ▶ implementazione (traduzione in codice eseguibile)

## Esempio: carrello

Consideriamo un carrello  $C$  che si muove su una rotaia.

Esso deve prima andare a sinistra, dove lo carica il robot  $R_1$ , e poi a destra, dove  $R_2$  lo scarica, e quindi ripetere le operazioni.



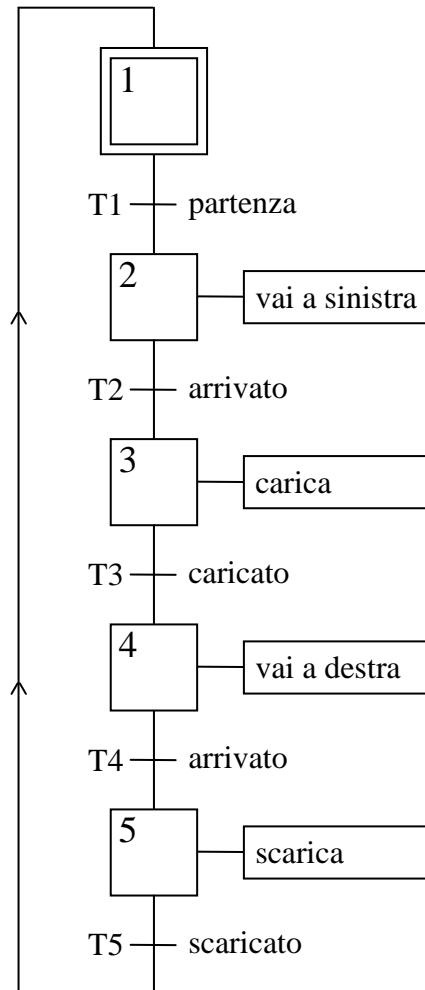
Sono disponibili due sensori di finecorsa:

- ▶  $FINE\_SX$
- ▶  $FINE\_DX$

e due comandi di movimento:

- ▶  $C\_SX$
- ▶  $C\_DX$

## Livello funzionale



La struttura principale del modello è una sequenza comando-misura-...-comando-misura:

- ▶ prima si dà un comando
- ▶ poi si verifica che esso sia andato a buon fine

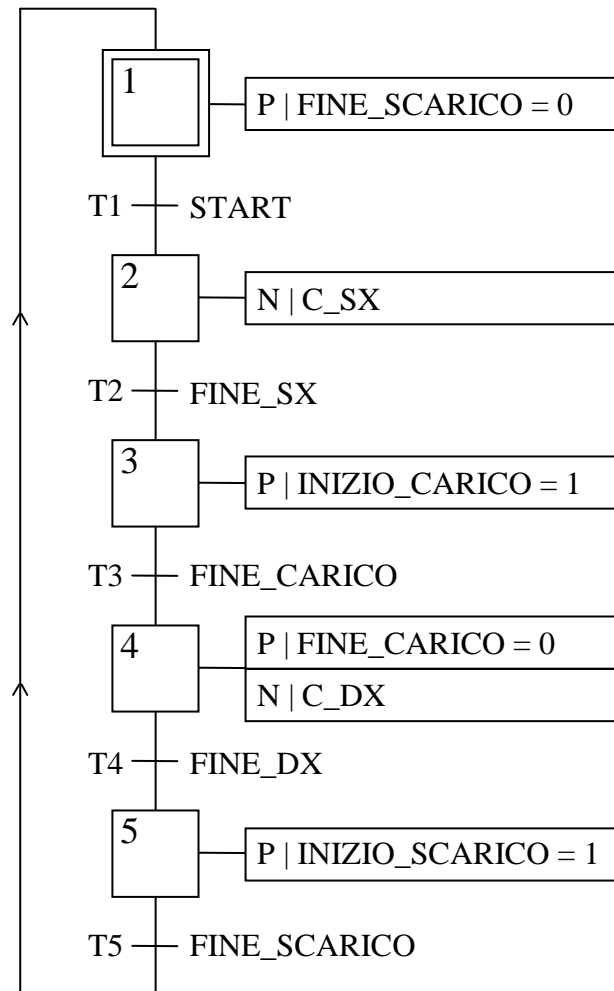
Al termine della sequenza si ritorna nello stato iniziale, di attesa (le sequenze produttive sono generalmente ripetitive).

Scrivendo il codice in questo modo si assume che ogni operazione vada sempre a buon fine.

Cosa succede se:

- ▶ la condizione a valle della fase attiva non si verifica mai?
- ▶ si verifica una condizione diversa da quella a valle della fase attiva?

## Livello operativo (1<sup>a</sup> parte): supervisore



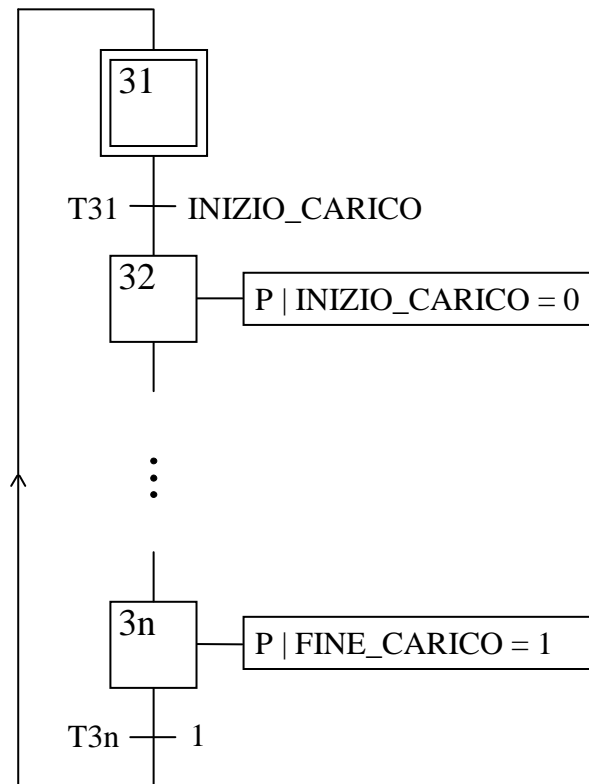
- ▶ START – condizione logica associata alla pressione di un pulsante da parte dell’operatore, oppure alla disponibilità di prodotti
- ▶ C\_SX – comando (continuo) che arriva direttamente ai motori del carrello per muoverlo verso sinistra
- ▶ FINE\_SX – sensore di finecorsa a sinistra (vale 1 se il sensore vede il carrello)
- ▶ ecc.

Le operazioni di carico e scarico vanno dettagliate ad un livello successivo, per esempio con un altro SFC. Gli eventi misurati associati a variabili interne devono essere consumati una volta sola.

Occorre resettare le variabili associate:

- ▶ FINE\_CARICO
- ▶ FINE\_SCARICO

## Livello operativo (2<sup>a</sup> parte): controllore del carico



La condizione associata alla prima transizione è espressa in termini della stessa variabile assegnata al passo 3 nello schema 1.

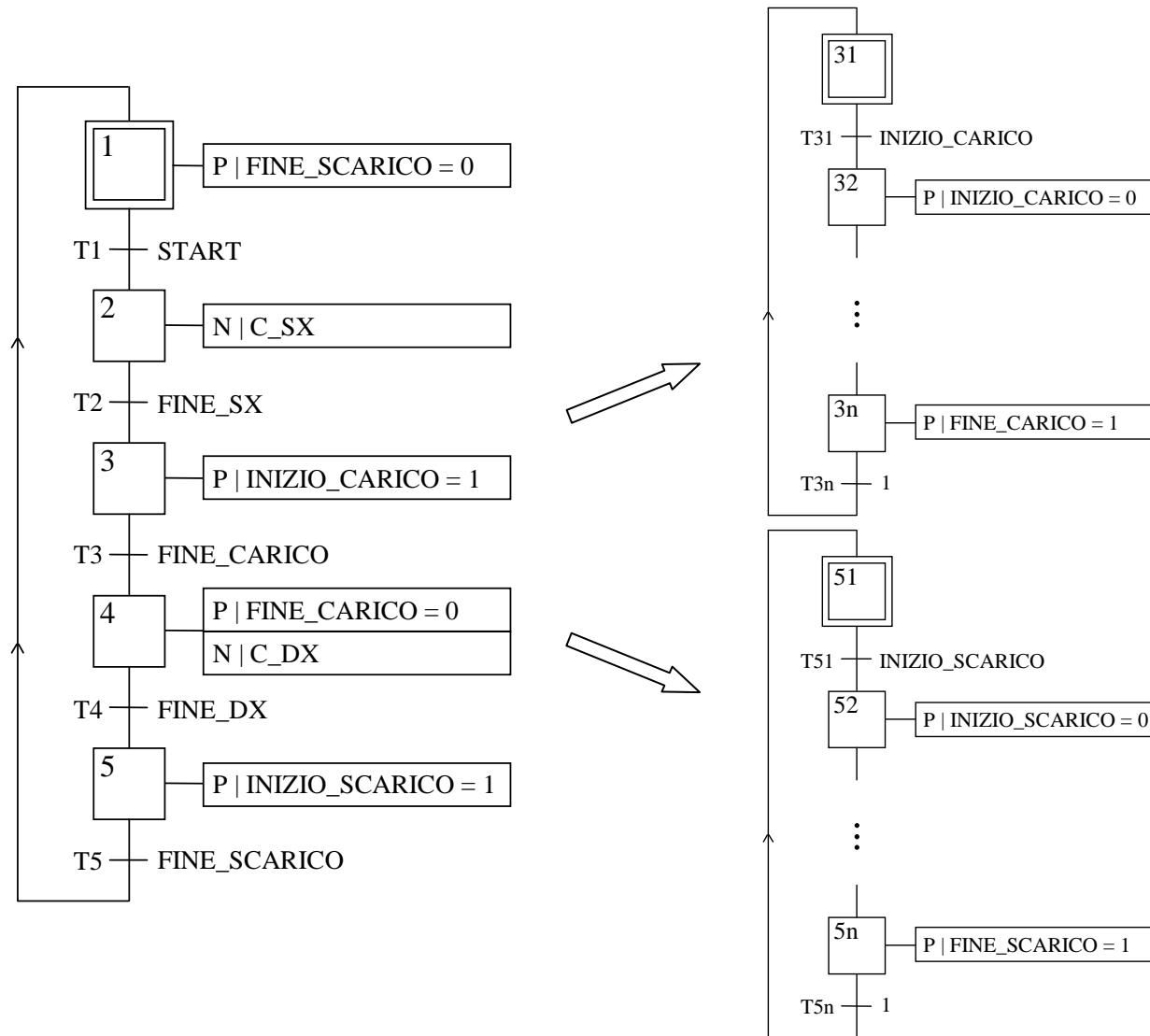
Al termine della procedura di carico si pone a 1 (TRUE) una variabile di FINE\_CARICO per segnalare l'avvenuto completamento dell'operazione.

La variabile è identica a quella che appare nella condizione associata alla transizione T3 immediatamente successiva al passo 3 nello schema 1: quando viene posta a 1, lo schema 1 (che era rimasto in attesa) evolve, attivando il passo 4.

Infine, la transizione T3n è superata automaticamente (ripristino delle condizioni che consentono l'avvio di un'altra operazione di carico).

Il controllore per l'operazione scarico è sviluppato in modo analogo.

## Schema complessivo (gerarchia di controllo)

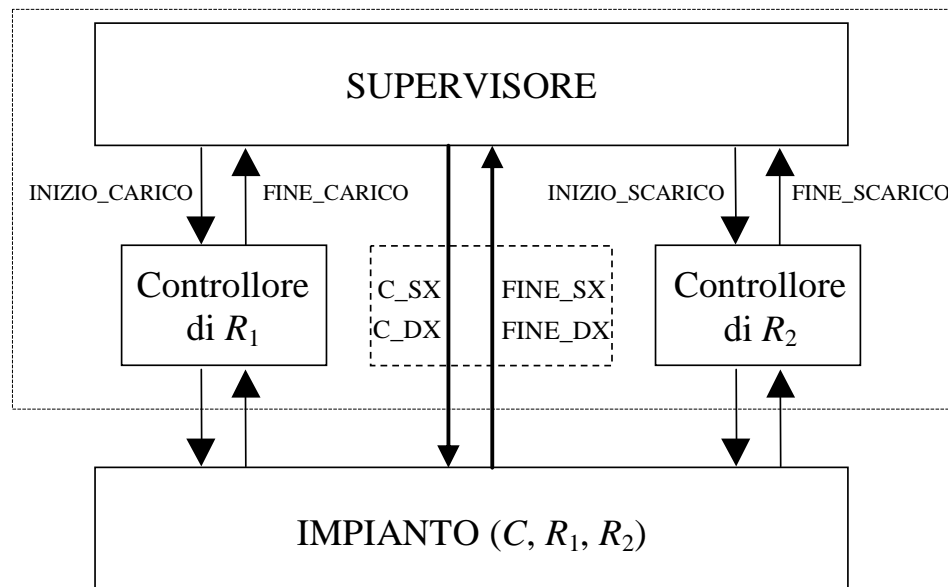


Il supervisore governa in modo gerarchico due controllori per il carico e lo scarico.

La sincronizzazione tra i 3 moduli si poteva realizzare alternatively usando

- le variabili di stato (v. dopo)
- le macrofasi





Il sistema di controllo è strutturato in 3 moduli:

- ▶ supervisore
- ▶ controllore di  $R_1$
- ▶ controllore di  $R_2$

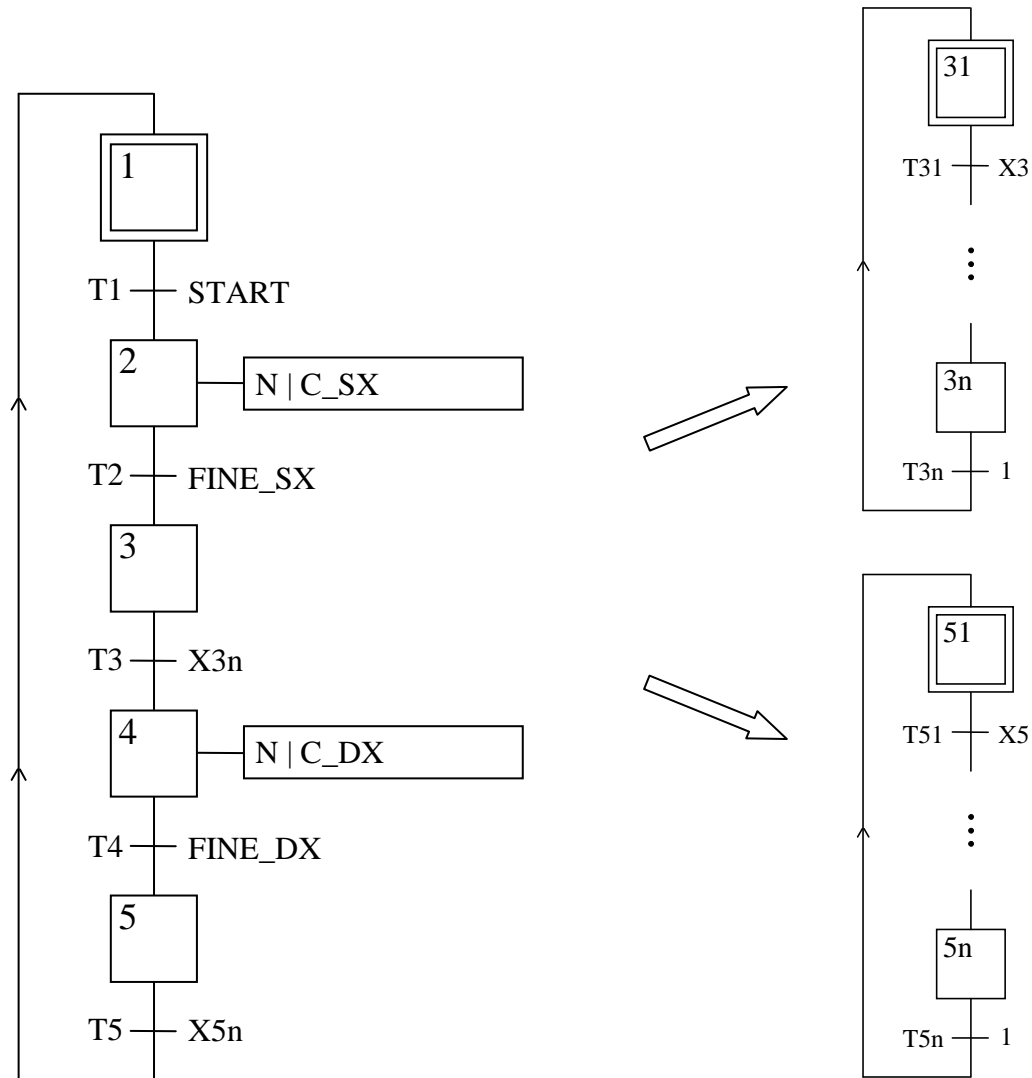
Il supervisore emette dei comandi che vanno direttamente verso l'impianto (segnali verso gli attuatori), **C\_SX** e **C\_DX**.

Analogamente, l'impianto risponde direttamente al supervisore con i segnali di misura, **FINE\_SX** e **FINE\_DX**.

Il supervisore coordina anche i due moduli di controllo generando i comandi **INIZIO\_CARICO** e **INIZIO\_SCARICO** (e riceve da essi i segnali **FINE\_CARICO** e **FINE\_SCARICO**, simili a misure).

I due controllori poi comunicheranno direttamente con l'impianto tramite altri segnali (v. motori e sensori dei robot).

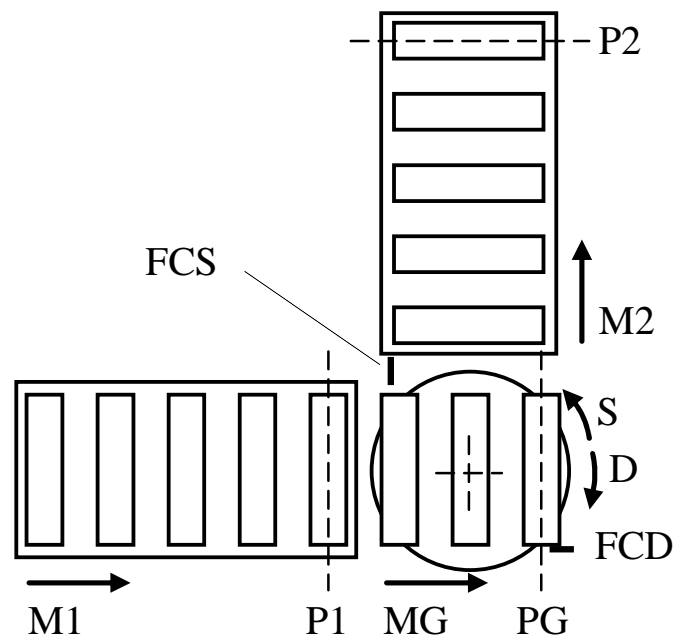
## Modalità di sincronizzazione basata sulle variabili di stato:



## Esempio: girola automatica

Consideriamo la girola automatica per il trasferimento pezzi tra due nastri trasportatori a rulli.

I pezzi, che sono depositi casualmente sul primo nastro, devono essere trasferiti, tramite la girola, sul secondo nastro, da dove vengono poi evacuati.



Segnali di ingresso (binari):

- ▶  $P_x$  presenza pezzo alla fine del nastro  $N_x$  ( $x = 1, 2$ )
- ▶  $PG$  presenza pezzo alla fine dei rulli della girola  $G$
- ▶  $FCS$  finecorsa rotazione a sinistra della girola  $G$
- ▶  $FCD$  finecorsa rotazione a destra della girola  $G$

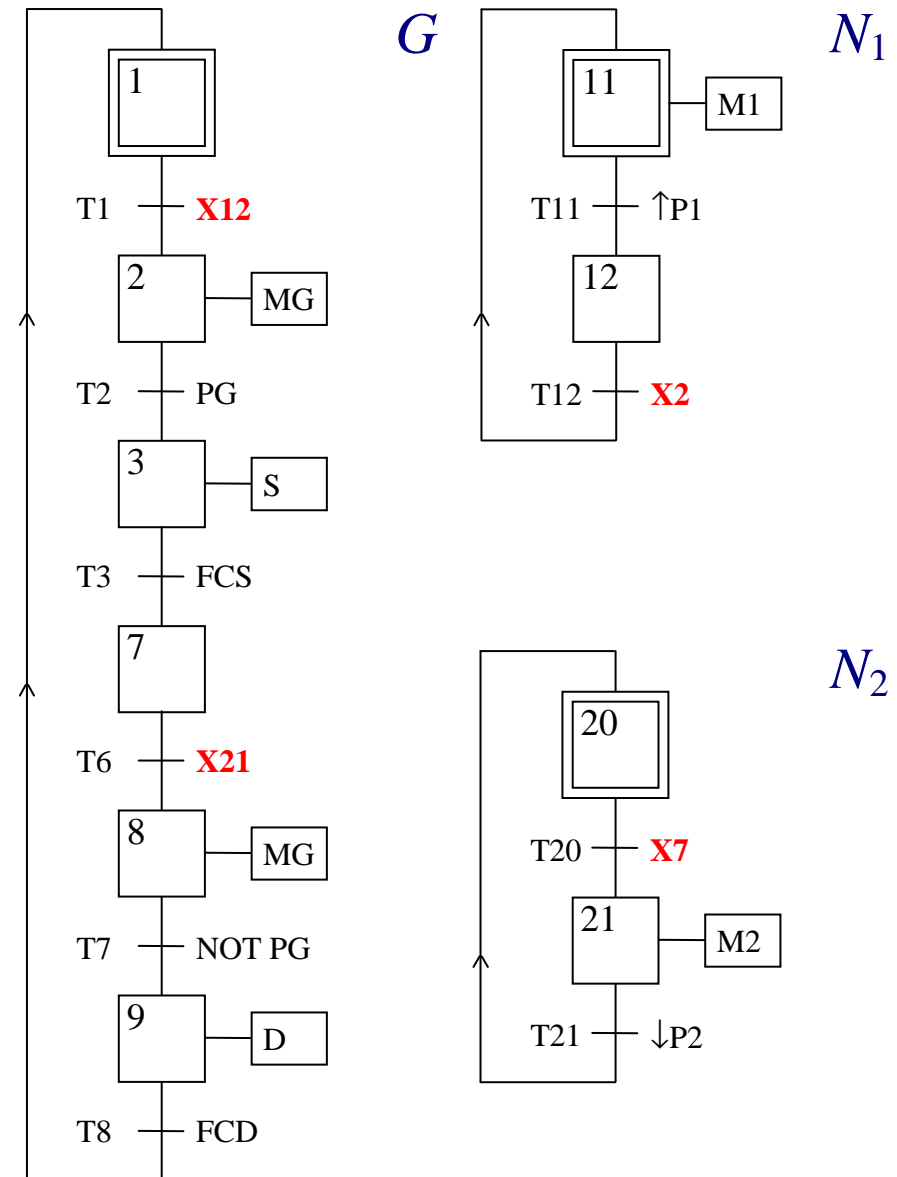
Segnali di comando (binari):

- ▶  $M_x$  motore del nastro  $N_x$  ( $x = 1, 2$ )
- ▶  $MG$  motore dei rulli della girola  $G$
- ▶  $S$  rotazione a sinistra della girola  $G$
- ▶  $D$  rotazione a destra della girola  $G$

Per descrivere la specifica funzionale del comportamento del sistema conviene osservare che i 3 dispositivi (i due nastri e la girola) devono funzionare in parallelo, opportunamente sincronizzati per consentire lo scambio dei pezzi.

Sfruttando il fatto che la sincronizzazione si può effettuare su eventi associati a variabili interne possiamo scomporre funzionalmente il problema in 3 moduli di controllo, uno per ciascun dispositivo.

In questo caso non c'è una vera e propria gerarchia tra i vari schemi, che si possono ritenere al medesimo livello.



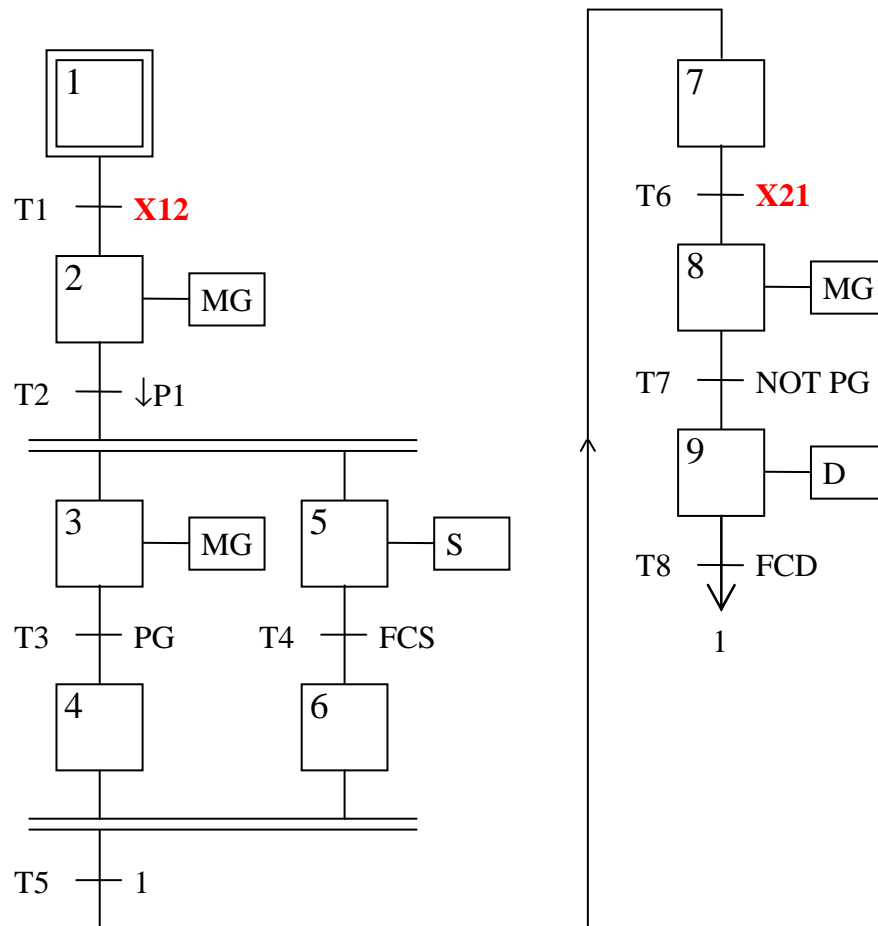
## Interpretazione del modello complessivo:

- ❶ Inizialmente,  $G$  e  $N_2$  sono in attesa di  $N_1$ , che è in movimento.
- ❷ Quando un pezzo viene a trovarsi su  $P1$ ,  $N_1$  si ferma (fase 12).
- ❸ A questo punto si attiva la fase 2 e i rulli della girola  $G$  trascinano il pezzo sulla girola. Contestualmente, i rulli di  $N_1$  si riattivano per spingere il pezzo sulla girola.
- ❹ Il nastro  $N_1$  rimane attivo anche quando il sensore  $P1$  si disattiva (pezzo trasferito), finché non arriva il pezzo successivo ( $\uparrow P1$ ), ecc.

## Commenti:

- ▶ avremmo anche potuto realizzare un unico SFC con la descrizione dall'inizio alla fine del percorso di un pezzo nel sistema: funziona, ma...
- ▶ fare attenzione alla sincronizzazione basata sulle variabili di stato: se successivamente si modifica lo schema, cambia la numerazione degli stati e i riferimenti si perdono! questo naturalmente non succede se si identificano gli stati con delle etichette testuali

## Modello della girola che sfrutta parallelismo tra avanzamento e rotazione:



Quando  $N_1$  è pronto per consegnare un pezzo (stato X12), i rulli della girola vengono attivati, per ricevere il pezzo.

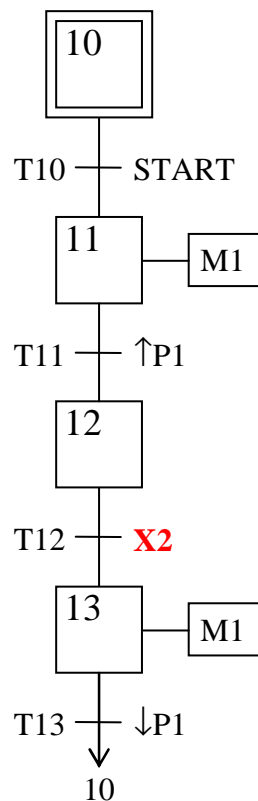
Quando il pezzo ha disimpegnato  $N_1$ , si inizia contestualmente a ruotare la girola, senza interrompere l'avanzamento del pezzo sulla girola.

Lo stato X7 indica che la girola è pronta per consegnare il pezzo a  $N_2$ .

Quando  $N_2$  è pronto per ricevere il pezzo (stato X21), i rulli della girola vengono attivati nuovamente fino a che il pezzo non disimpegna la girola, che poi viene successivamente ruotata nella posizione iniziale.

### Modello N<sub>1</sub> perfezionato:

- ▶ si introduce una condizione di START prima di qualunque altra azione
- ▶ si gestisce lo scarico del pezzo sulla girola



### Modello N<sub>2</sub> perfezionato:

- ▶ non si interrompe l'azione continua M2, se non è necessario (se la stessa azione è associata a due fasi successive, la transizione di fase non interrompe l'azione)

