



# Corso di Automazione industriale

Lezione 6

PLC – SFC

Esercizi

# Esercizio 1

Si consideri un sistema di trasporto di pietre per mezzo di un carrello.

L'operatore determina l'inizio del ciclo tramite un apposito pulsante START. Il carrello percorre per intero il binario da sinistra a destra e si arresta in attesa di essere caricato. Le pietre, dopo essersi accumulate in un serbatoio, vengono meccanicamente trasferite nel carrello, il quale deve automaticamente muoversi lungo il binario da destra a sinistra.

# Esercizio 1

Come **INPUT** vi sono a disposizione sei sensori:

**S** START

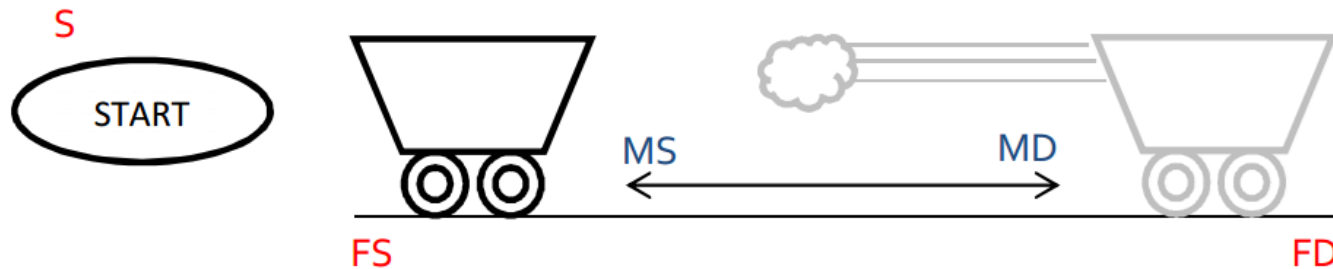
**FS** Fondo binario sx

**FD** Fondo binario dx

**SV** Serbatoio vuoto

**FSG** Fondo serbatoio giù

**FSS** Fondo serbatoio su



Come **OUTPUT** troviamo:

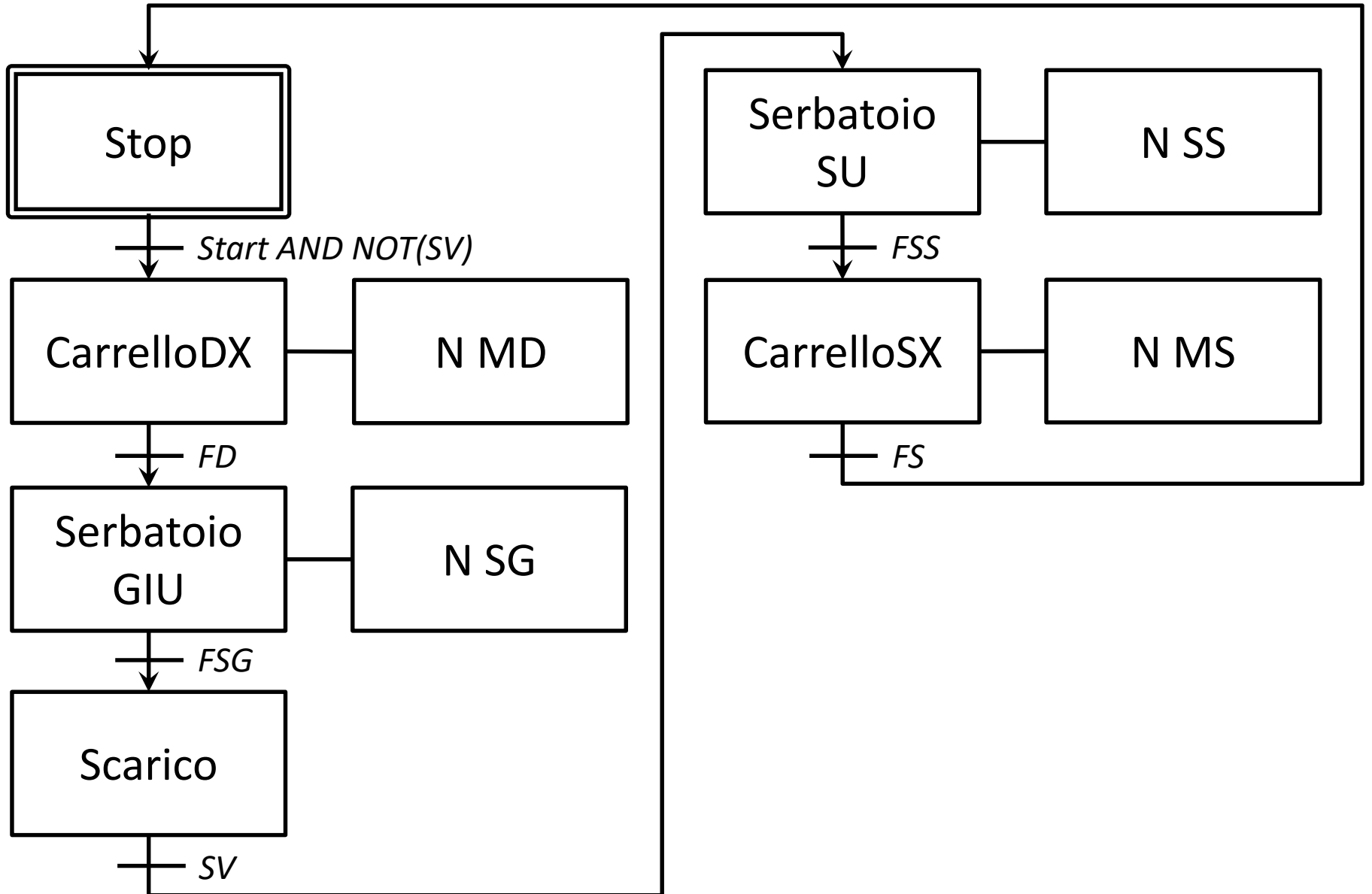
**MD** Motore carrello verso dx

**MS** Motore carrello verso sx

**SG** Motore serbatoio in giù

**SS** Motore serbatoio in su

# Esercizio 1



# Esercizio 1.1

Aggiungiamo all'esercizio precedente una fermata per manutenzione ogni 100 cicli.

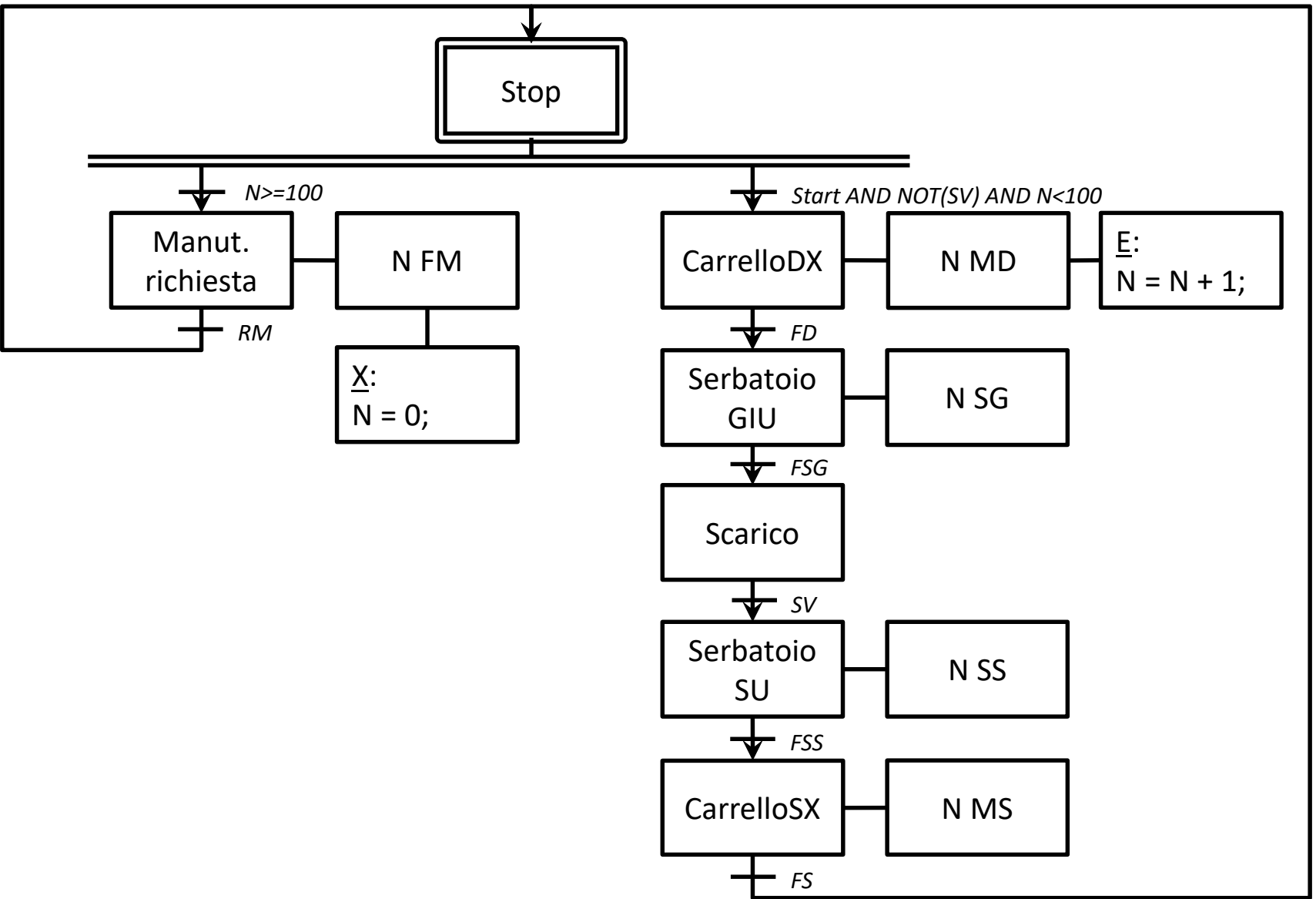
Serve aggiungere:

FM (Fermo Manutenzione) come output

RM (Reset Manutenzione) come input

N.B.: Servirà anche creare una variabile contatore!

# Esercizio 1.1



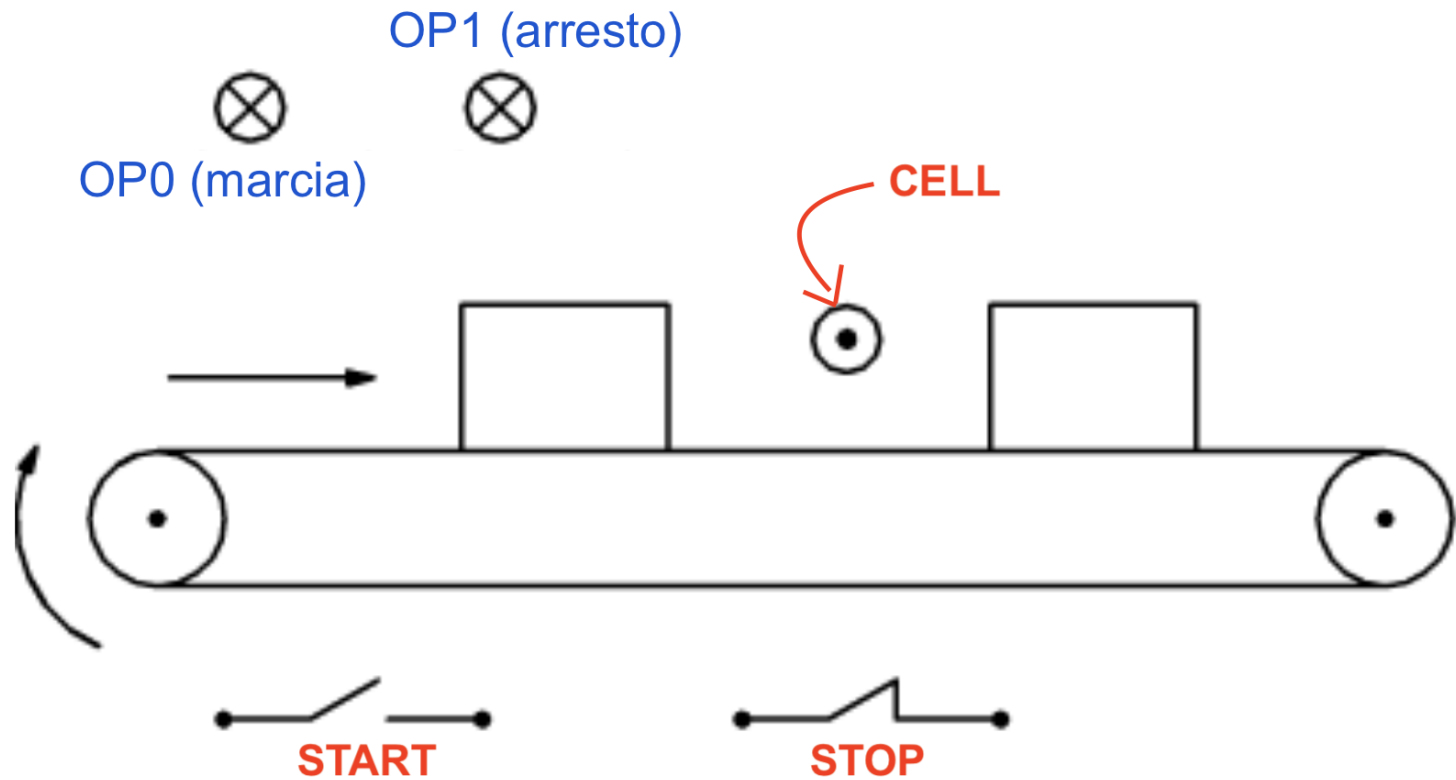
## Esercizio 2

Si consideri un sistema conta-pezzi con nastro trasportatore. Il nastro viene azionato da un motore, comandato da due pulsanti START e STOP. Lo stato di marcia e/o di arresto deve essere mostrato tramite due lampade distinte.

Ciascun pezzo viene posizionato all'inizio del nastro e, ad ogni passaggio di un pezzo davanti alla fotocellula, deve essere gestito il conteggio. Il sistema si deve fermare in automatico ogni 50 pezzi.

Il nastro può essere fermato manualmente in qualunque momento premendo il tasto STOP. Il riavvio può avvenire tramite il tasto START, ma il conteggio deve riprendere da dove si era fermato.

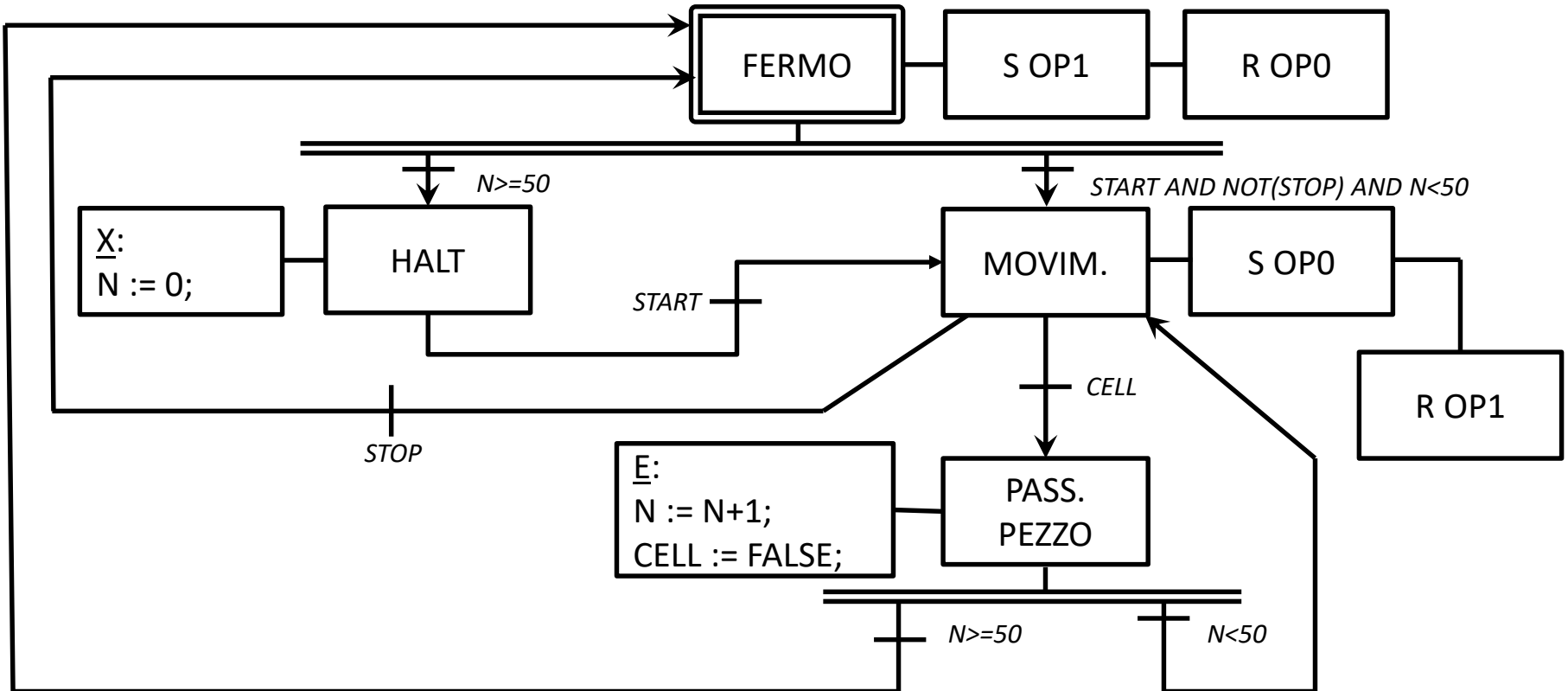
# Esercizio 2



## Esercizio 2

- All'avvio, il motore che aziona il nastro deve essere fermo, quindi dovrà essere accesa solamente *OP1*.
- Premendo il tasto *START*, il motore si avvia: si deve accendere *OP0* e si deve spegnere *OP1*.
- Ogni volta che il motore è in funzione e si rileva un pezzo davanti alla fotocellula *CELL*, deve essere incrementato il conteggio.
- Quando il conteggio raggiunge il valore 50, il nastro deve essere fermato: si deve accendere *OP1* e si deve spegnere *OP0* (*ad un successivo riavvio il conteggio riparte da 0*).
- Fermando il nastro prima del valore limite il conteggio deve riprendere dal valore a cui era arrivato.

# Esercizio 2



# Esercizio 3

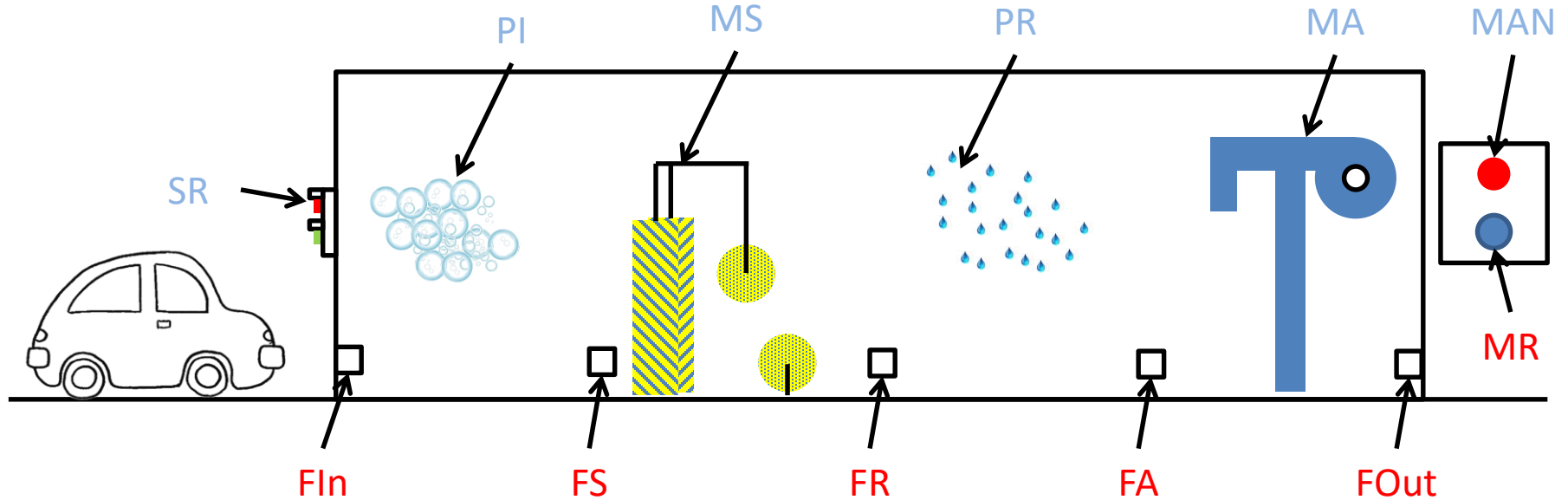
Si consideri un sistema di lavaggio automatico per autovetture.

Il cliente si avvicina al nastro movimentatore quando il semaforo è verde. Le fasi di lavaggio sono: insaponatura, spazzolatura, risciacquo e asciugatura.

Tutte le fasi sono precedute da una fotocellula che rileva l'arrivo della vettura in quella sezione dell'impianto.

Ogni 1000 lavaggi l'autolavaggio deve bloccarsi in attesa di manutenzione effettuata da un operatore.

# Esercizio 3



## Input

## Output

**Fin** Fotocellula in ingresso

**SR** Semaforo stop (0=VERDE, 1=ROSSO)

**FS** Fotocellula spazzolatura

**PI** Pompa insaponatura

**FR** Fotocellula risciacquo

**MS** Motore spazzolatura

**FA** Fotocellula asciugatura

**PR** Pompa risciacquo

**FOut** Fotocellula uscita

**MA** Motore asciugatura

**MR** Manutenzione reset (anche out)

**MAN** Stop per manutenzione

# Esercizio 3

L'impianto può essere visto come un insieme di singoli impianti:

- Insaponatura
- Spazzolatura
- Risciacquo
- Asciugatura

Ognuno di questi «impianti» deve avviarsi quando la fotocellula davanti ad essi si attiva e spegnersi quando la fotocellula dopo essi si disattiva.

# Esercizio 3

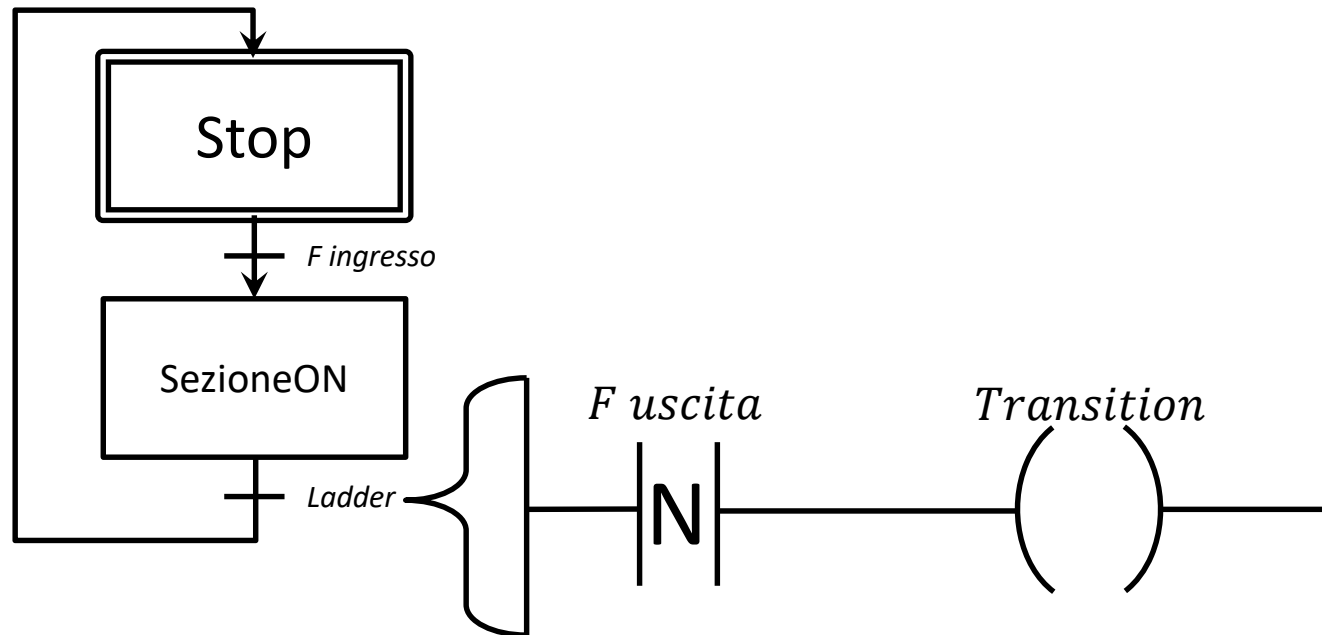
Come mostrato nella lezione 4, la soluzione a questo esercizio prevede una gestione «distribuita» di ogni singola parte dell'impianto.

In SFC non abbiamo la possibilità di gestire più «cicli di esecuzione» in un singolo chart, per questo motivo è necessario utilizzare più di un programma.

Avremo quindi un programma per ogni parte dell'impianto (nel chart che gestisce l'insaponatura si gestiranno anche il semaforo e la manutenzione).

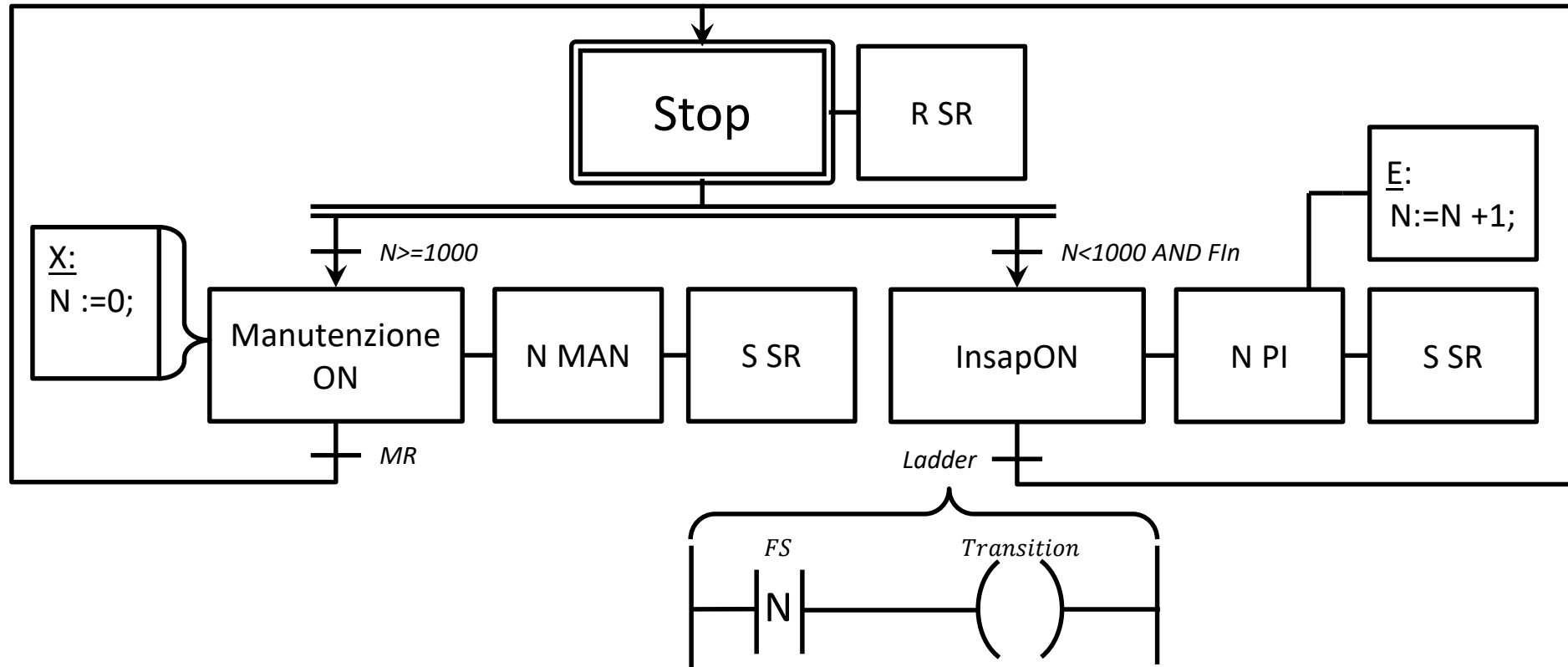
# Esercizio 3

Ogni parte dell'impianto avrà il seguente SFC:



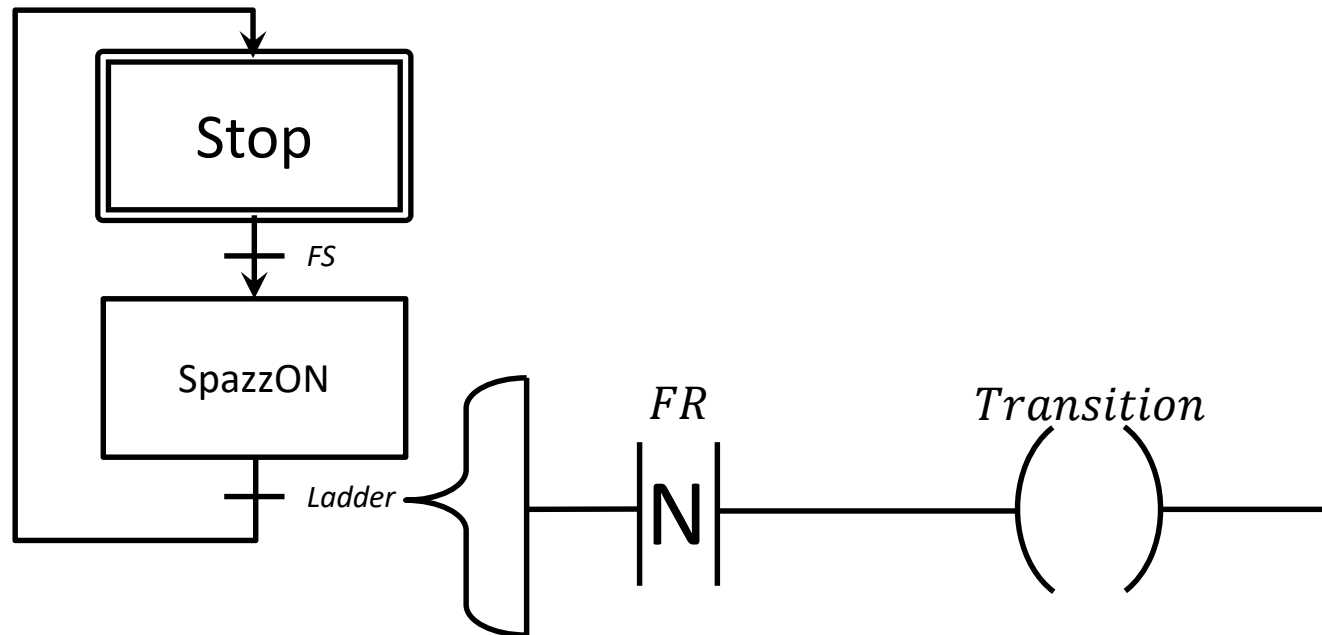
# Esercizio 3

Oltre a questi programmi, nel programma di gestione dell'insaponatura, è necessario introdurre la gestione di semaforo e manutenzione.



# Esercizio 3

Esempio di SFC per la sezione si spazzolatura



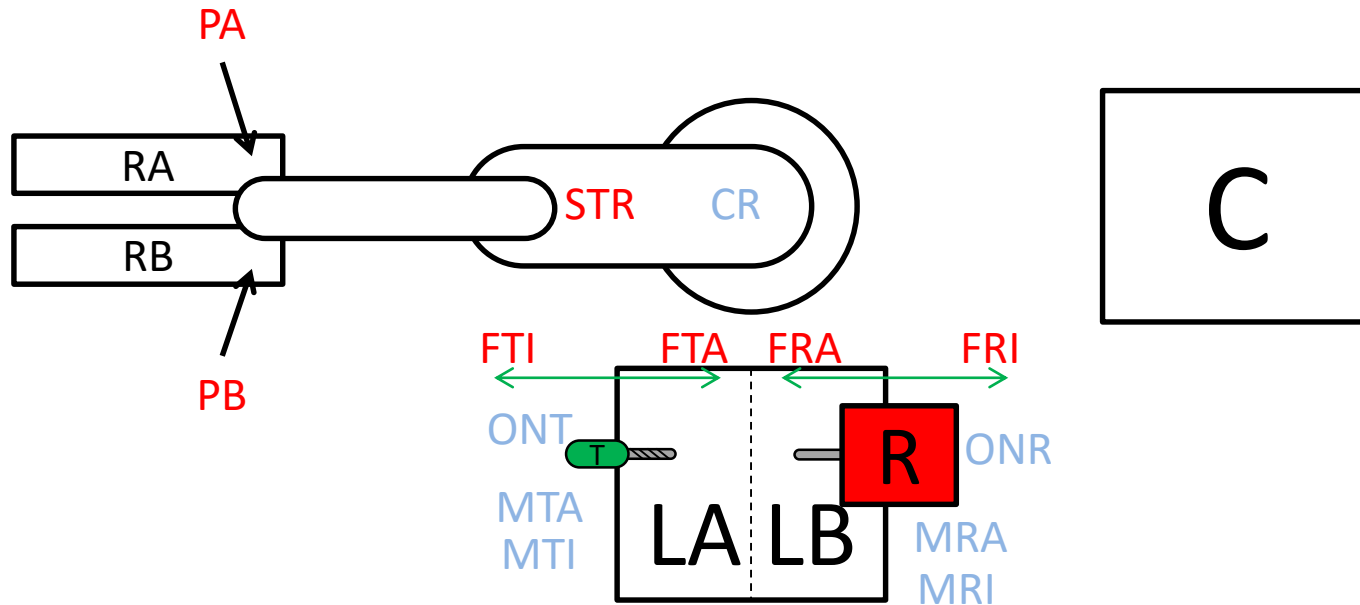
# Esercizio 4

Si consideri un sistema di foratura e rivettatura automatica per lamiera.

All'arrivo dei due pezzi un robot effettua il prelievo dei componenti (in successione uno all'altro) e li posiziona nella maschera di montaggio. Al termine del posizionamento avviene la foratura per mezzo di trapano automatico (5 sec) e la rivettatura (10 sec).

Al termine della lavorazione il robot sposta il pezzo unito in un contenitore.

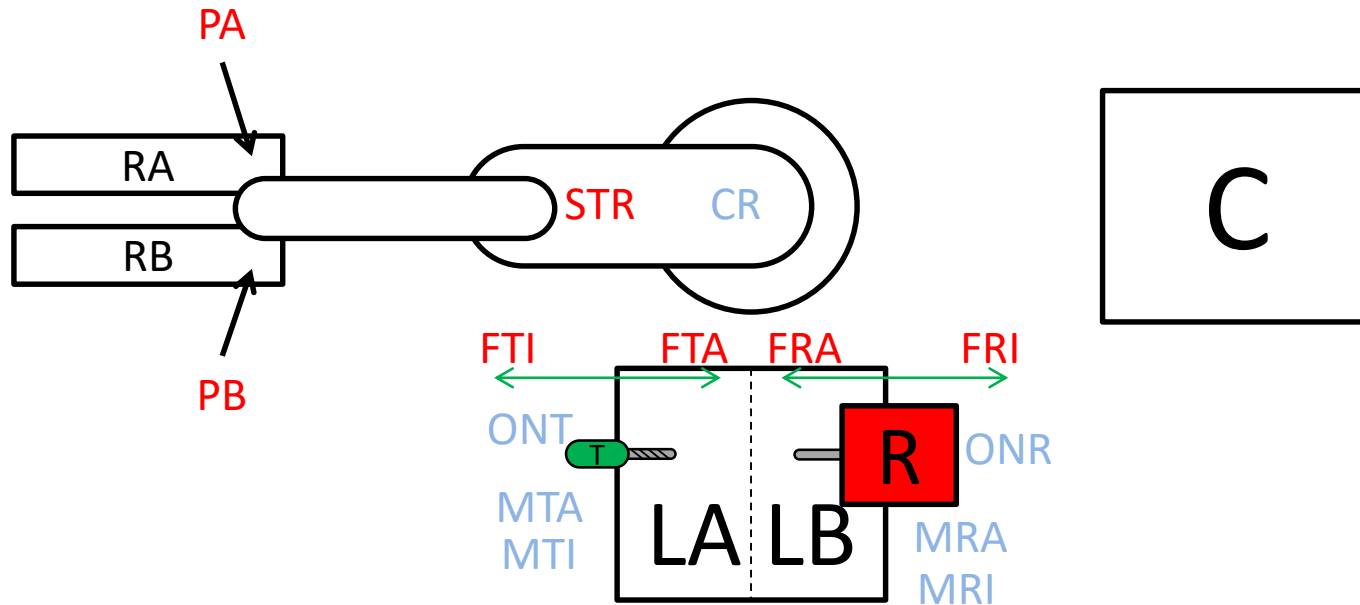
# Esercizio 4



## Input

- |            |                                    |            |                                 |
|------------|------------------------------------|------------|---------------------------------|
| <b>PA</b>  | Sensore presenza nastro A          | <b>FTA</b> | Finecorsa trapano avanti        |
| <b>PB</b>  | Sensore presenza nastro B          | <b>FRI</b> | Finecorsa rivettatrice indietro |
| <b>STR</b> | Stato robot (0=FINE, 1=ESECUZIONE) | <b>FRA</b> | Finecorsa rivettatrice avanti   |
| <b>FTI</b> | Finecorsa trapano indietro         |            |                                 |

# Esercizio 4



## Output

CR	Comando robot (0=STOP, 1=prelievo RA deposito LA, 2=prelievo RB deposito LB, 3=prelievo L deposito C)	ONT	On trapano
MTA	Movimento avanti trapano	MRA	Movimento avanti rivettatrice
MTI	Movimento indietro trapano	MRI	Movimento indietro rivettatrice
		ONR	On rivettatrice

# Esercizio 4

Ragioniamo sui passi da seguire per realizzare un prodotto finito:

- 1)Attendere che PA e PB vadano a 1
- 2)Comando 1 al robot e attendere la fine dell'esecuzione
- 3)Comando 2 al robot e attendere la fine dell'esecuzione
- 4)Muovere il trapano avanti fino a raggiungere il suo finecorsa avanti
- 5)Azionare il trapano per 5 secondi
- 6)Muovere il trapano indietro fino a raggiungere il suo finecorsa indietro
- 7)Muovere la rivettatrice avanti fino a raggiungere il suo finecorsa avanti
- 8)Azionare la rivettatrice per 10 secondi
- 9)Muovere la rivettatrice indietro fino a raggiungere il suo finecorsa indietro
- 10)Comando 3 al robot e attendere la fine dell'esecuzione
- 11)Comando 0 al robot

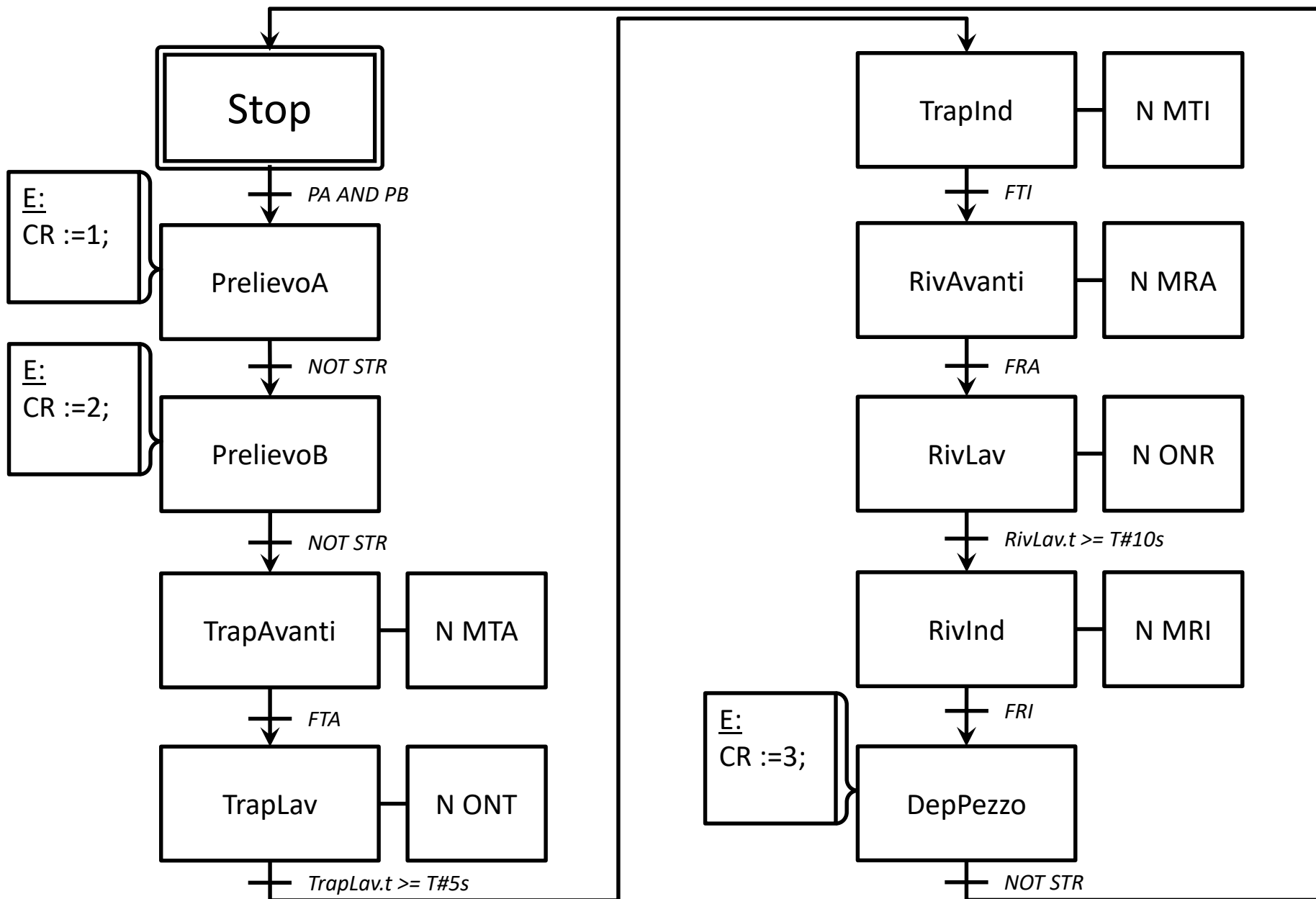
# Esercizio 4

Questo esempio, che nella lezione 4 è stato «risolto» utilizzando il ladder con degli stati calza a pennello con la logica SFC.

Abbiamo una serie di stati che si susseguono in maniera regolare, con delle semplici transizioni tra uno e l'altro.

La soluzione è «banale», gli stati sono i punti descritti nella slide 17.

# Esercizio 4



# Conclusioni

## Considerazioni SFC

È un linguaggio adatto a sviluppare algoritmi di controllo logico con stati finiti.

Non sempre è adatto a controlli logici di tipo «decentralizzato» (abbiamo un'esecuzione singola).

Come vedremo nell'ultima parte del corso si sposa perfettamente con la logica delle reti di Petri.