

SFC

Pagina Iniziale: [Automazione Industriale](#)

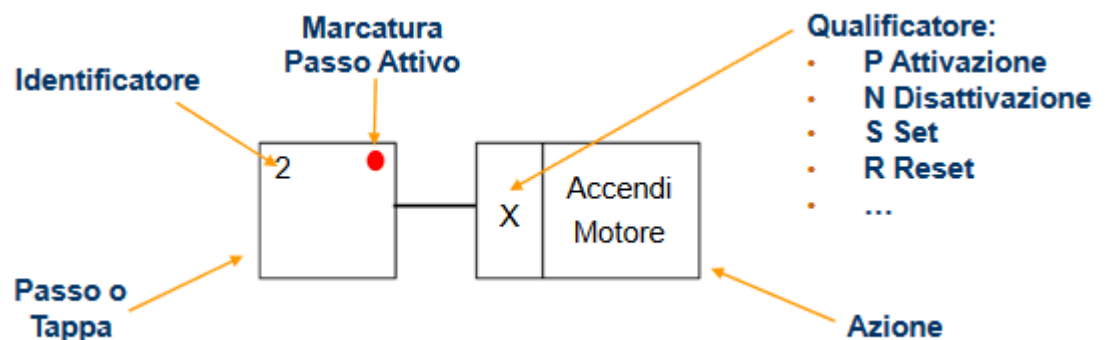
Indice: [Index](#)

Sequential Function Chart

Nasce dalla necessità di usare modelli formali per rappresentare funzioni di controllo logico per automazione industriale, per standardizzare formalismi rappresentativi, per arricchire gli automi (non hanno parallelismo), semplificano le reti di Petri e renderle adatte ad applicazioni industriali e per permettere una facile interpretazione ai dispositivi di controllo

Elementi di base

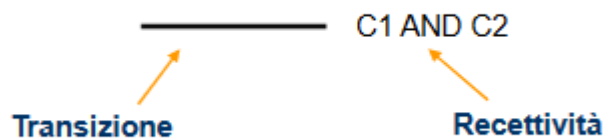
- Passo: stato di esecuzione della sequenza, possiede un identificatore, può essere attivo o passivo, può avere una serie di azioni associate che vengono eseguite quando il passo è attivo e possono possedere un qualificatore per specificare momento/periodo di esecuzione, può essere collegato **SOLO** a transizioni



✓ Qualificatori:

| | | |
|-----------|-------------------------|---|
| N | Non-Store | L'azione termina quando il passo diventa inattivo |
| S | Set (Stored) | L'azione continua anche quando il passo diventa inattivo, e termina quando l'azione viene resettata |
| R | Reset | Termina un'azione attivata con i qualificatori S, SD, SL o DS |
| L | Time Limited | L'azione comincia quando il passo diventa attivo e continua finché il passo diventa inattivo o trascorre un certo intervallo di tempo |
| D | Time Delayed | Un timer viene settato quando il passo diventa attivo; se il passo è ancora attivo dopo l'azzeramento del timer, l'azione comincia e termina quando il passo si disattiva |
| P | Pulse | L'azione comincia quando il passo diventa attivo/disattivo e viene eseguita una sola volta |
| SD | Stored and time Delayed | L'azione comincia dopo un ritardo anche se il passo diventa inattivo e continua finché non resettata |
| DS | Delayed and Stored | Un timer viene settato quando il passo diventa attivo; se il passo è ancora attivo dopo l'azzeramento del timer, l'azione comincia e continua finché non resettata |
| SL | Stored and time Limited | L'azione comincia quando il passo diventa attivo e continua finché non viene resettata o non trascorre un certo intervallo di tempo |

- Transizione: passaggio tra un passo e l'altro durante l'esecuzione, possiedono un identificatore, possono essere abilitate o disabilitate, possono scattare e sono associate alla reattività (Condizione logica su ingressi) possono essere collegate **SOLO** a passi



- Azione: ogni transizione ha associata una o più azioni
- Condizioni logiche: controllano lo scatto delle transizioni e quindi l'evoluzione dell'esecuzione (Vero = scatto Falso = fermo)
- Arco orientato: identifica il passaggio tra un passo e l'altro tramite una transizione controllata da una condizione logica

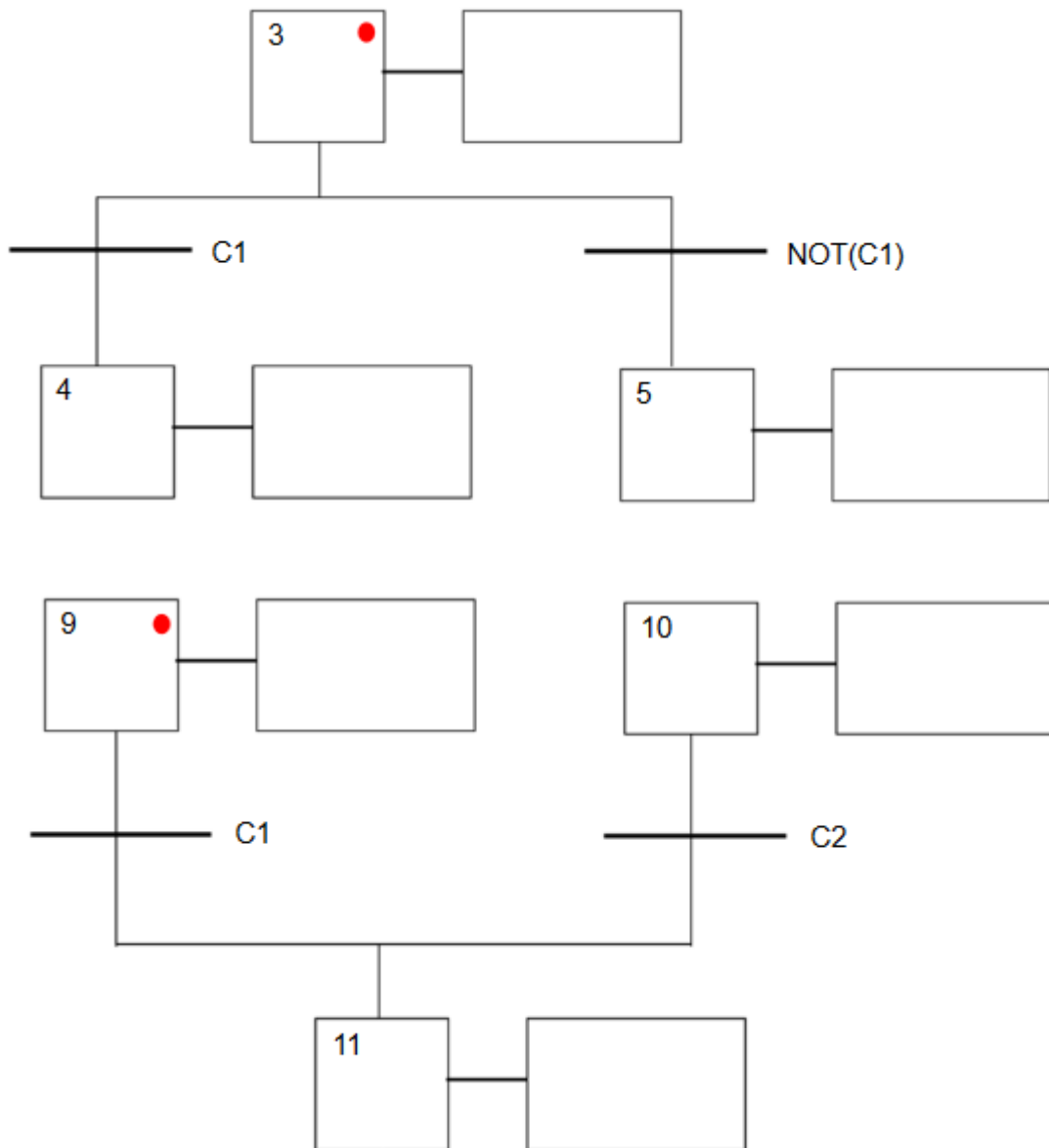
Operatori base

Sono rappresentabili due caratteristiche basilari per sistemi ad eventi:

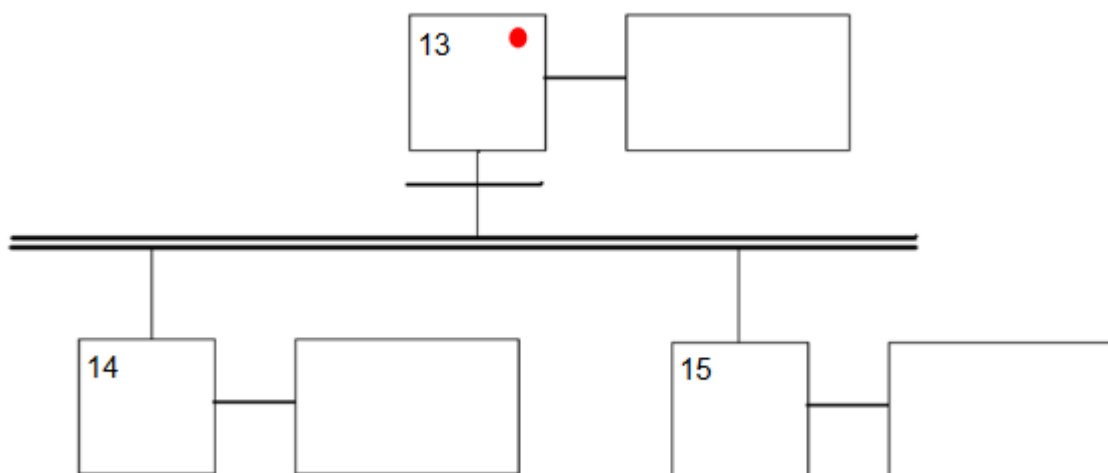
- Scelta (operatori di tipo OR)
- Parallelismo (operatori di tipo AND)

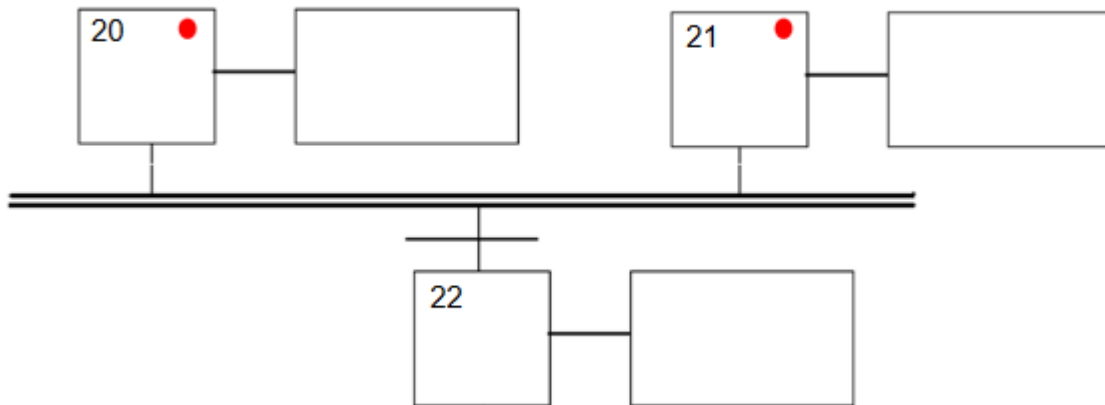
A tali caratteristiche corrispondono operatori grafici diversi caratterizzati da un inizio (divergenza) ed una fine (convergenza)

OR



AND





Regole di evoluzione

Una transizione si dice **abilitata** se tutti i passi a monte sono attivi

Una transizione si dice **superabile** se è abilitata e la sua recettività assume il valore logico vero (la transizione può scattare)

Regola 1:

se una transizione è superabile essa viene effettivamente superata: tutti i passi a monte vengono disattivati mentre tutti quelli a valle vengono abilitati

Regola 2:

tutte le transizioni superabili in un certo istante vengono superate contemporaneamente

Note:

- In un OR si cerca di evitare di avere situazioni ambigue sulle transizioni tramite mutua esclusione o dando priorità ad una delle due transizioni
- Cercare di evitare di avere situazioni in cui un passo si deve attivare e disattivare contemporaneamente (Convenzionalmente rimane attivo)

Interpretazione di un SFC

La rappresentazione formale delle regole di evoluzione (o interpretazione) di uno schema SFC si chiama algoritmo di evoluzione (o interpretazione)

Esistono molte varianti per gli algoritmi di interpretazione poiché esistono interpretazioni diverse di uno schema

Algoritmi di Evoluzione senza ricerca di stabilità

Rappresentazione in forma algoritmica delle regole di evoluzione tali che in presenza di sequenze di transizioni con condizione logica vera, queste vengono superate in cicli diversi, e non nello stesso ciclo

Tutte le uscite associate ai passi intermedi vengono assegnate



✓ **Algoritmo di
evoluzione
con ricerca di
stabilità**

