



Corso di Automazione industriale

Lezione 5

PLC - SFC

Storia

Prima degli anni '60 il CONTROLLO SEQUENZIALE era visto come ESTENSIONE DEL CONTROLLO DI TIPO CONTINUO o al più DIGITALE.

A partire dagli anni '60 si sviluppa la teoria sugli AUTOMI A STATI FINITI, i cui modelli formali permettono un'ANALISI MATEMATICA approfondita, ma scarsamente utile ai fini della PROGETTAZIONE degli algoritmi.

Negli anni '70 la progettazione dei sistemi di automazione è abbastanza elementare, si basa su RAPPRESENTAZIONI CIRCUITALI o su DESCRIZIONI TESTUALI.

...

Storia

...

Nel 1975 in Francia viene istituita una commissione per FORMALIZZARE uno strumento di PROGETTAZIONE di tipo DESCRITTIVO orientato al CONTROLLO SEQUENZIALE. Nasce GRAFCET.

Negli anni '80 GRAFCET viene recepito con il nome di SEQUENTIAL FUNCTIONAL CHART nello standard IEC (Comitato Elettrotecnico Internazionale) 848.

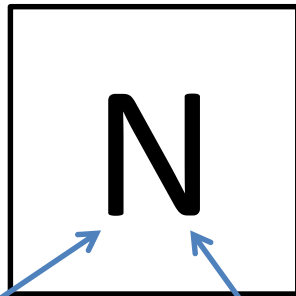
Elementi base

Stato (fase)

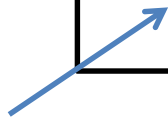
Definizione

Lo STATO è una precisa CONDIZIONE OPERATIVA di una parte del sistema complesso.

Rappresentazione grafica

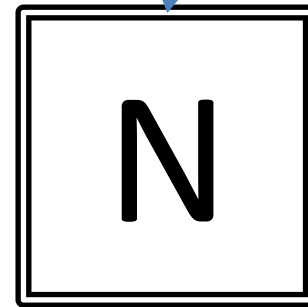


Numero stato



... o meglio Nome univoco

Stato iniziale con doppio contorno



Elementi base

Stato (fase)

Osservazioni

Lo STATO è una condizione INVARIANTE del sistema in esame, che può modificarsi o essere modificata soltanto in seguito ad un EVENTO.

Durante l'evoluzione dinamica del sistema complesso, un qualsiasi STATO di una qualsiasi parte del sistema può trovarsi solo in due possibili condizioni: ATTIVO o INATTIVO.

Elementi base

Stato (fase)

Sintassi

IL PLC definisce per ogni stato due variabili:

- MARKER, ovvero la variabile booleana che dice se è attivo oppure no
- TIMER, cioè la variabile interna che rappresenta la durata di attivazione (se lo stato è attivo)

Solitamente

- Variabile MARKER: NomeStato.X
- Variabile TIMER: NomeStato.T

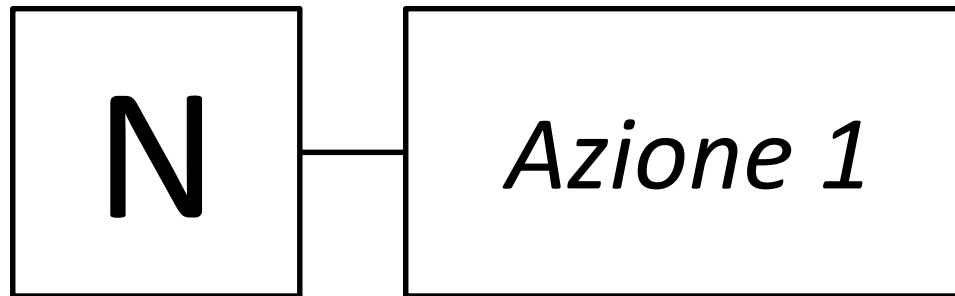
Elementi base

Azione

Definizione

Tutte le OPERAZIONI eseguite dal sistema in esame quando si trova in una precisa CONDIZIONE OPERATIVA (STATO) vengono chiamate AZIONI.

Rappresentazione grafica



Elementi base

Azione

Osservazioni

Ogni STATO del sistema in esame è composto da un insieme determinato di AZIONI.

Dal punto di vista del SISTEMA DI CONTROLLO che si occupa di eseguire il PROGRAMMA di controllo LOGICO-SEQUENZIALE, una AZIONE corrisponde ad una o più PROCEDURE, ovvero da un INSIEME DI ISTRUZIONI che vengono eseguite in maniera seriale fintanto che lo STATO è ATTIVO, ovvero fino a quando un EVENTO non cambia lo STATO.

Elementi base

Azione

Sintassi

IL PLC definisce per ogni azione tre variabili:

- A_m , ovvero l'identificatore univoco dell'azione
- Q_m , che definisce la tipologia di azione
- V_m , la variabile booleana che rappresenta lo stato dell'azione

N.B.: In molti ambienti di sviluppo le azioni (e anche le transizioni) possono essere definite con uno dei linguaggi IEC 61131

Elementi base

Azione

Sintassi

Tipologia di qualificatori

- N: Normal non stored

Se l'azione A_n è di tipo N verrà eseguita ad ogni ciclo del PLC se lo stato relativo ad essa è attivo

- P: Pulse

Se l'azione A_n è di tipo P verrà eseguita solo il primo ciclo di attivazione dello stato relativo ad essa

Elementi base

Azione

Sintassi

Tipologia di qualificatori

- S: Set

Se l'azione A_n è di tipo S verrà eseguita ad ogni ciclo del PLC finché la medesima azione A_n (in uno stato successivo) avrà qualificatore R

- R: Reset

Se l'azione A_n è di tipo R (come già anticipato) termina l'esecuzione di un'azione precedentemente abilitata con un qualificatore di tipo S

Elementi base

Azione

Sintassi

Tipologia di qualificatori

- L: Limited Time

Se l'azione A_n è di tipo L verrà eseguita ad ogni ciclo del PLC per $t\#Ts$ secondi a partire dall'attivazione dello stato

N.B.: se lo stato viene disabilitato l'azione termina

- D: Time Delayed

Se l'azione A_n è di tipo D verrà eseguita ad ogni ciclo del PLC dopo $t\#Ts$ secondi dall'attivazione dello stato

Elementi base

Azione

Sintassi

Tipologia di qualificatori

- SD: Stored / time Delayed

Se l'azione A_n è di tipo SD equivale ad una azione S ritardata di $t\#T_s$ secondi

- SL: Stored / time Limited

Se l'azione A_n è di tipo SL equivale ad una azione S, disabilitata dopo $t\#T_s$ secondi (a prescindere dallo stato attuale)

Elementi base

Transizione

Definizione(i)

Il PASSAGGIO da uno STATO PRECEDENTE ad uno STATO SUCCESSIVO a seguito di un evento, è detto TRANSIZIONE.

La verifica di tipo LOGICO che determina il verificarsi o meno di un EVENTO è detta CONDIZIONE.

Osservazioni

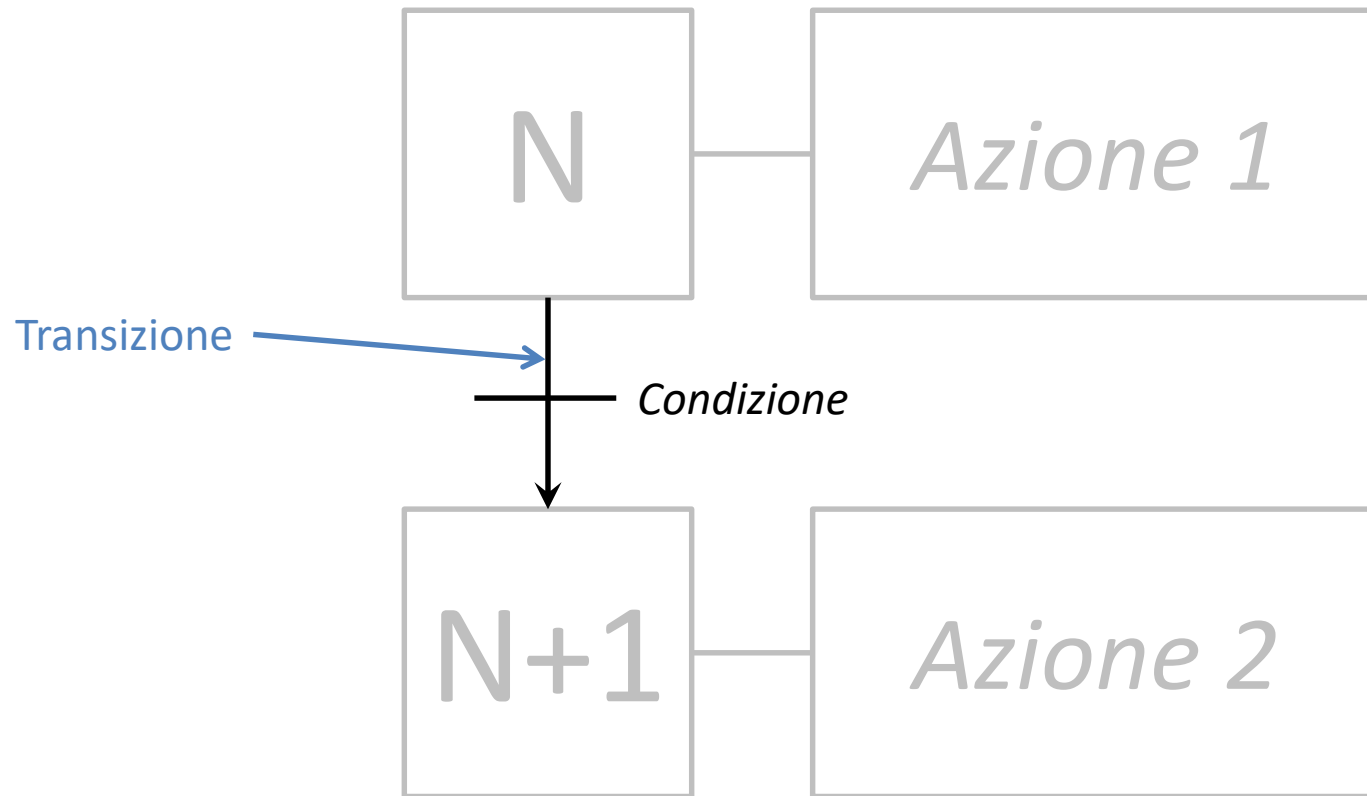
Ogni CONDIZIONE è associata ad una TRANSIZIONE.

Ogni CONDIZIONE è espressa sotto forma di una FUNZIONE LOGICA, ovvero tramite una ESPRESSIONE BOOLEANA che può essere VERA o FALSA.

Elementi base

Transizione

Rappresentazione grafica



Regole di evoluzione

Da ricordare

Regola 1

Una transizione si dice ABILITATA se tutte le fasi a monte sono attive.

Una transizione si dice SUPERABILE se è abilitata e la condizione ad essa associata assume il valore «vero».

Regole di evoluzione

Regola 2

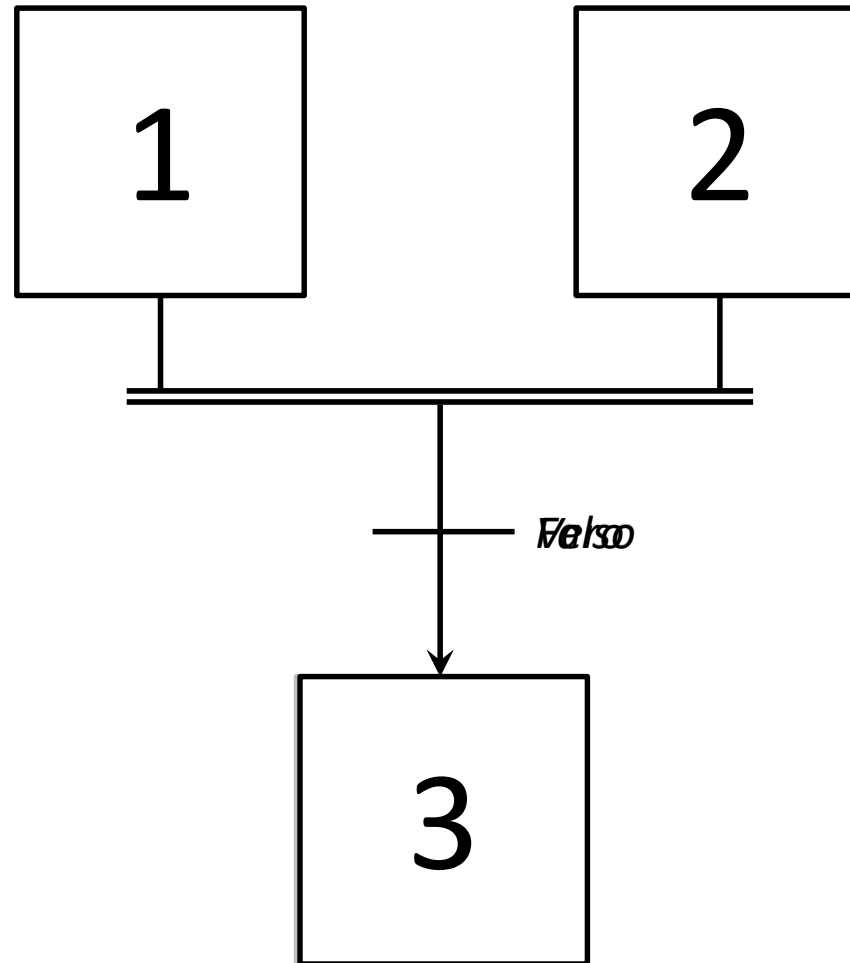
Quando una transizione è superabile essa viene superata: tutte le fasi a monte vengono disabilitate, tutte le fasi a valle vengono attivate

È molto semplice: quando la condizione diventa true si attivano le fasi sottostanti e si disattivano quelle sopra

N.B.: la corretta sintassi è sempre stato - transizione - stato - transizione - ...

Regole di evoluzione

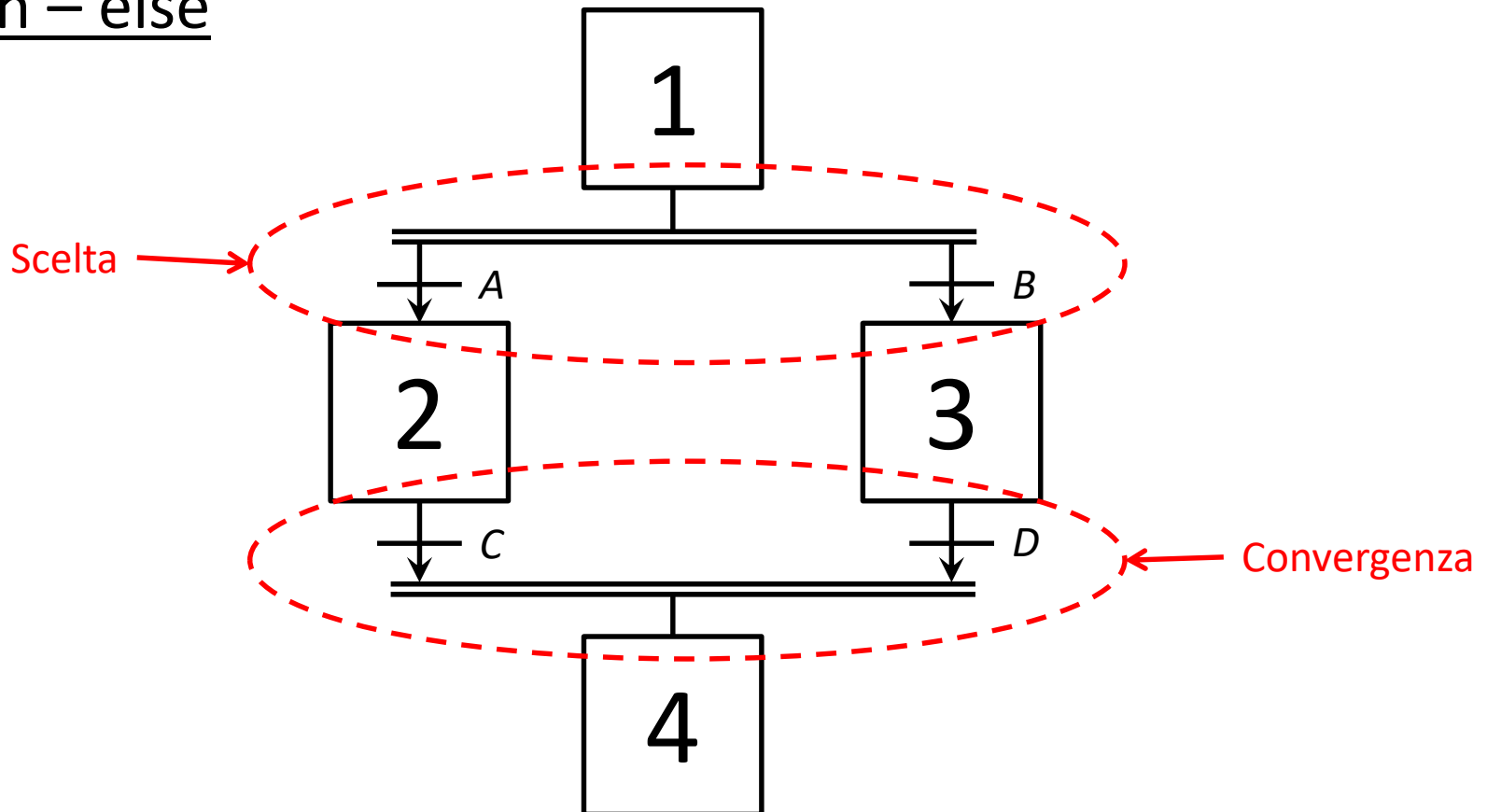
Esempio



Strutture base

Definite le regole viste in precedenza, le strutture base che possono essere create sono le seguenti:

If – then – else

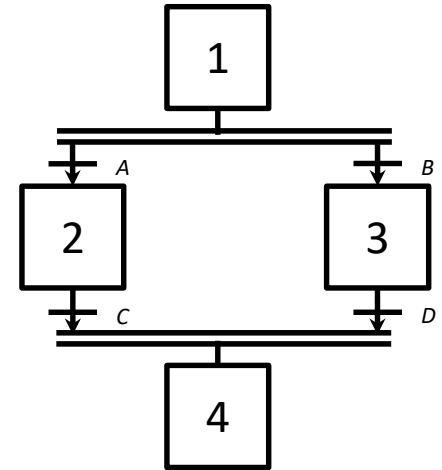


Strutture base

Osservazioni

Le condizioni A e B devono essere mutuamente esclusive.

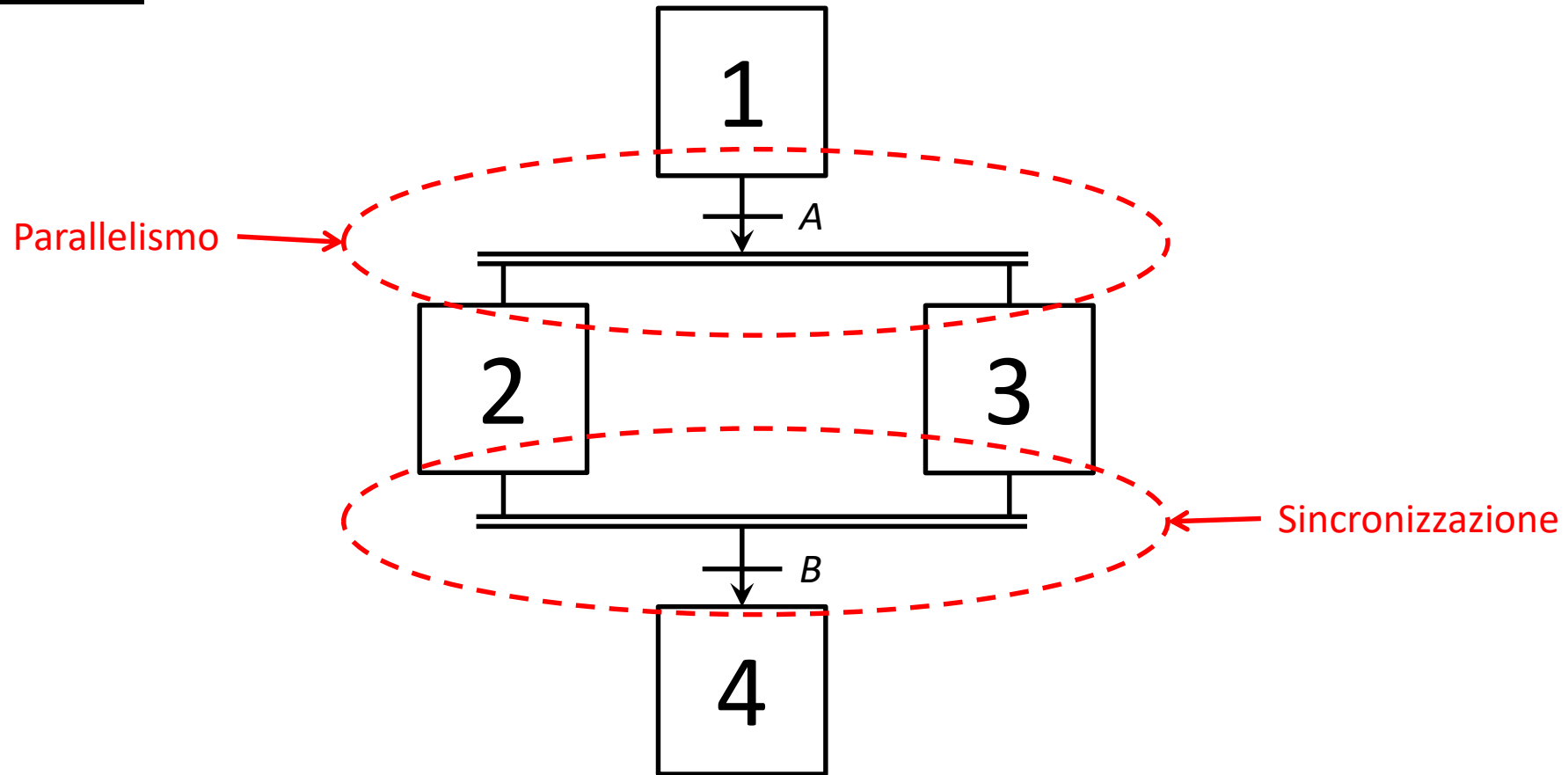
Se non lo fossero si rischierebbe di avere la fase 4 attivata «due volte».



N.B.: Come vedremo nell'ultima parte del corso questa è una differenza tra SFC e reti di Petri.

Strutture base

Parallelo

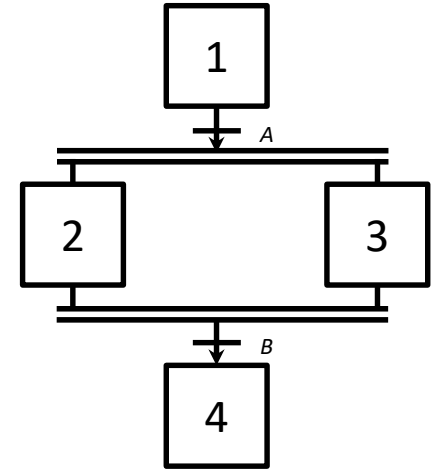


Strutture base

Osservazioni

Le fasi 2 e 3 sono attivate contemporaneamente.

Nel caso di presenza, in uno dei rami, di più di una fase, la verifica della condizione B avviene all'attivazione dell'ultima fase di tutti i rami.



SFC in un caso reale

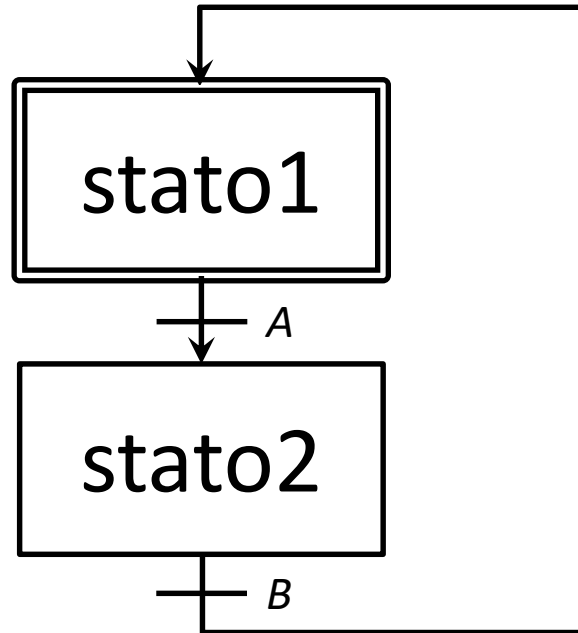
In molti ambienti di sviluppo (come anche nel caso di B&R Automation Studio) l'SFC viene integrato con gli altri linguaggi della IEC 61131.

Ciò significa che, ad ogni fase (chiamata «state»), è possibile associare l'esecuzione di una porzione di codice scritta in uno dei linguaggi IEC 61131.

Per questo motivo, non sono presenti tutte le tipologie di azioni mostrate in precedenza.

Esempio

Esempio base



Cosa succede se A e B sono entrambe true?

Il PLC continua a commutare ciclicamente tra stato 1 e stato 2

Esempio

Esempio azioni e if – then – else

Sviluppare un software che consenta, andando ad attivare un attuatore per 1 secondo, di scartare dei prodotti con peso inferiore ad una determinata soglia

Ingressi

FC: fotocellula

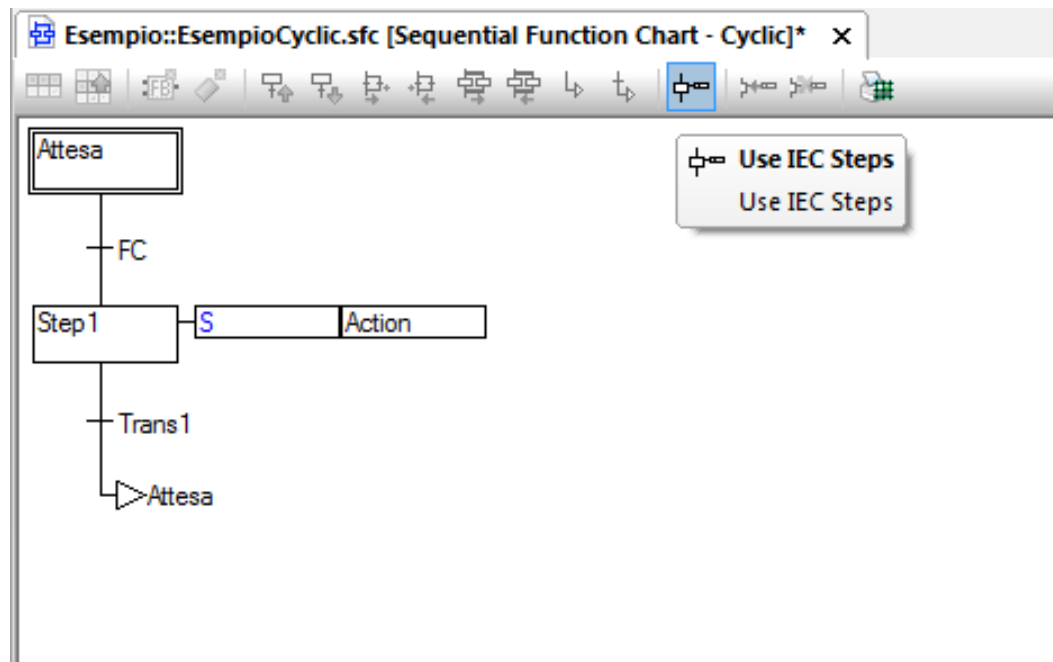
Peso: peso del prodotto

Uscite

Attuatore: attivazione dell'attuatore

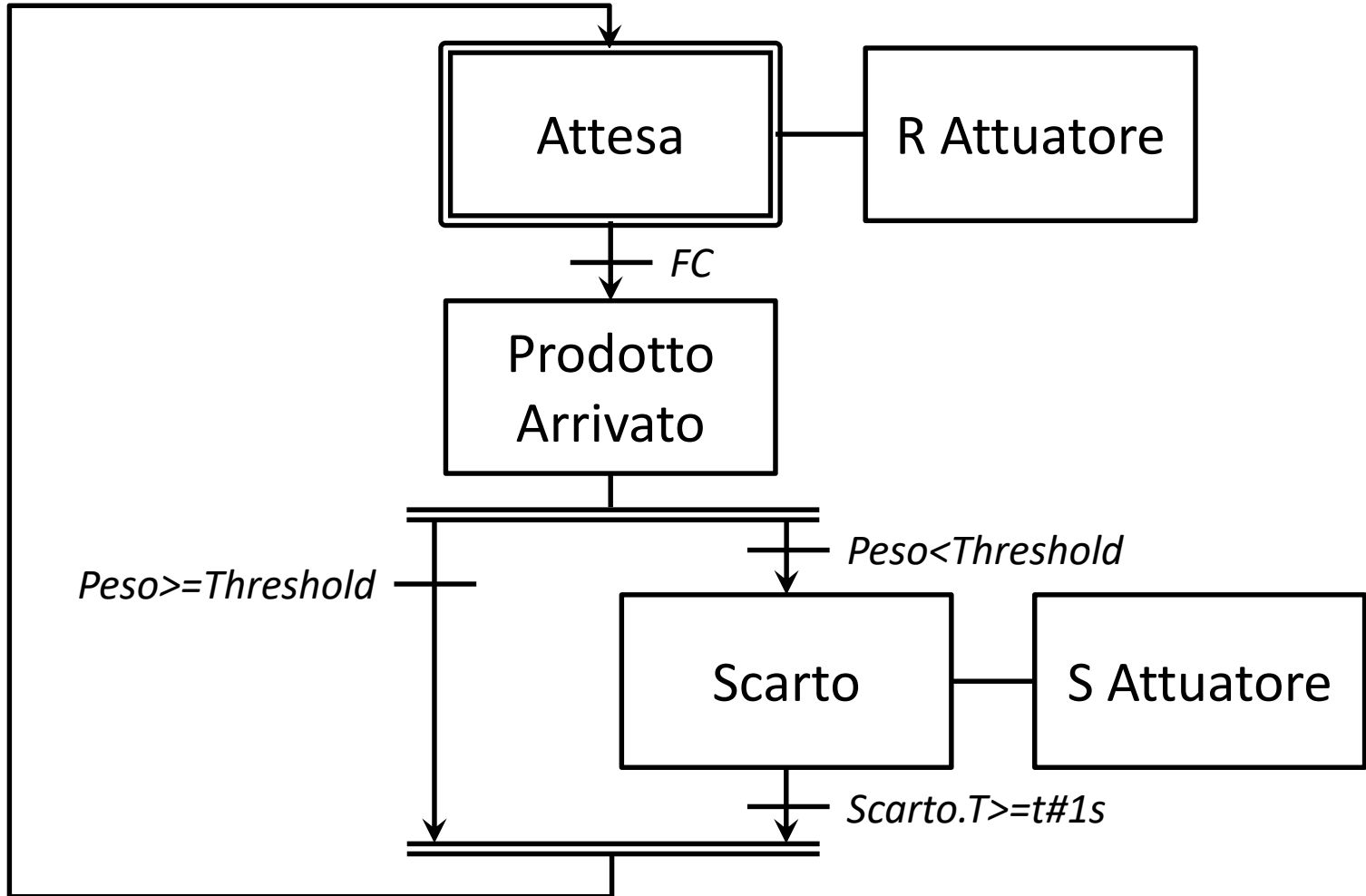
Esempio

N.B.: Con Automation Studio se vogliamo utilizzare la IEC per creare delle azioni è necessario cliccare sul pulsante Use IEC Steps come mostrato in figura.



Esempio

Esempio azioni e if – then – else



Esempio

Esempio stati con limiti temporali

Sviluppare un software che attivi l'elettrovalvola U se il sensore troppo-pieno STP resta attivo per più di 10 s e la mantiene attivata per 30 s

Ingressi

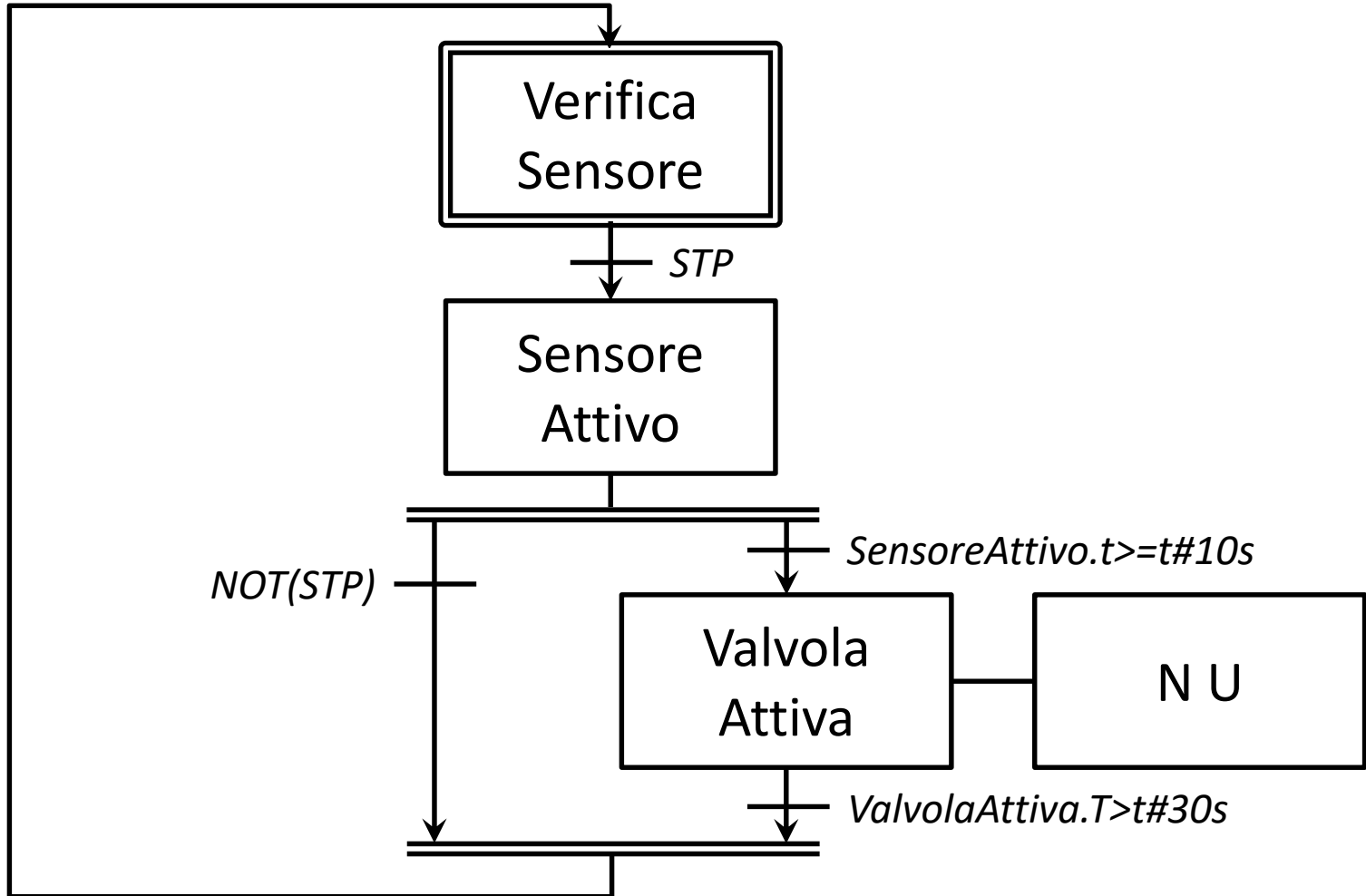
STP: sensore troppo-pieno

Uscite

U: attivazione elettrovalvola

Esempio

Esempio stati con limiti temporali



Esempio

Esempio azioni in altri linguaggi

Sviluppare un software che avvisa il passaggio del decimo cliente, utilizzando una fotocellula sull'ingresso

Ingressi

FC: fotocellula

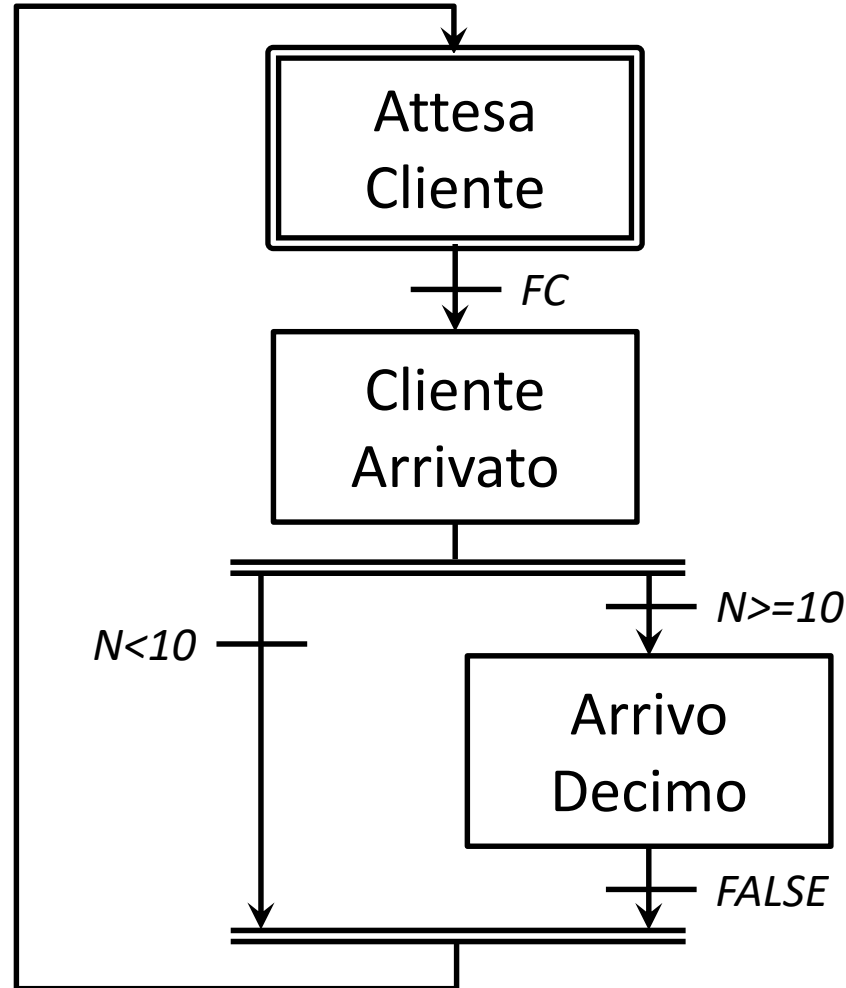
Uscite

Decimo: raggiungimento del decimo cliente

N.B.: durante gli esercizi, dove verrà chiesto di realizzare un regolatore in linguaggio SFC non sarà possibile utilizzare questa soluzione

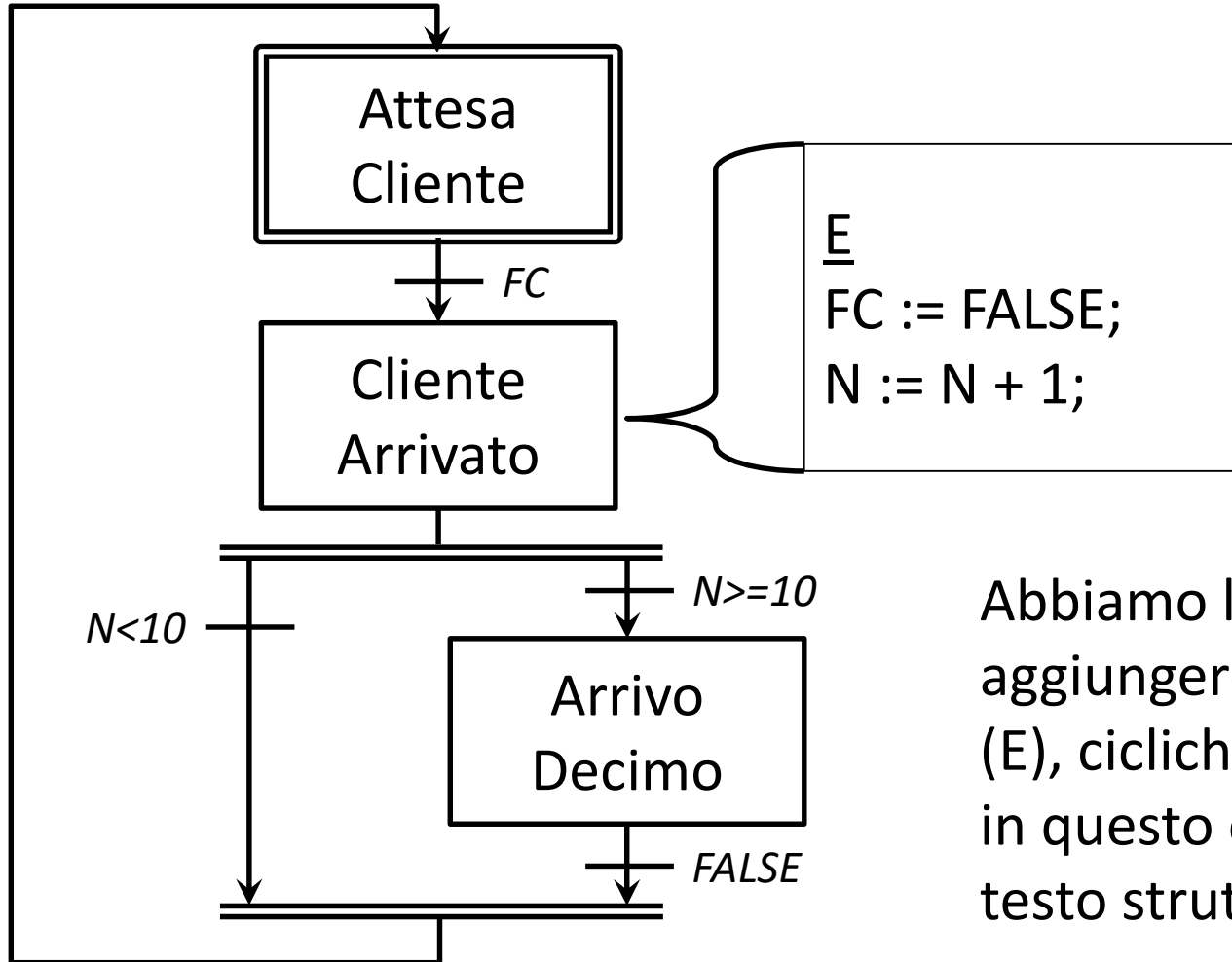
Esempio

Esempio azioni in altri linguaggi



Esempio

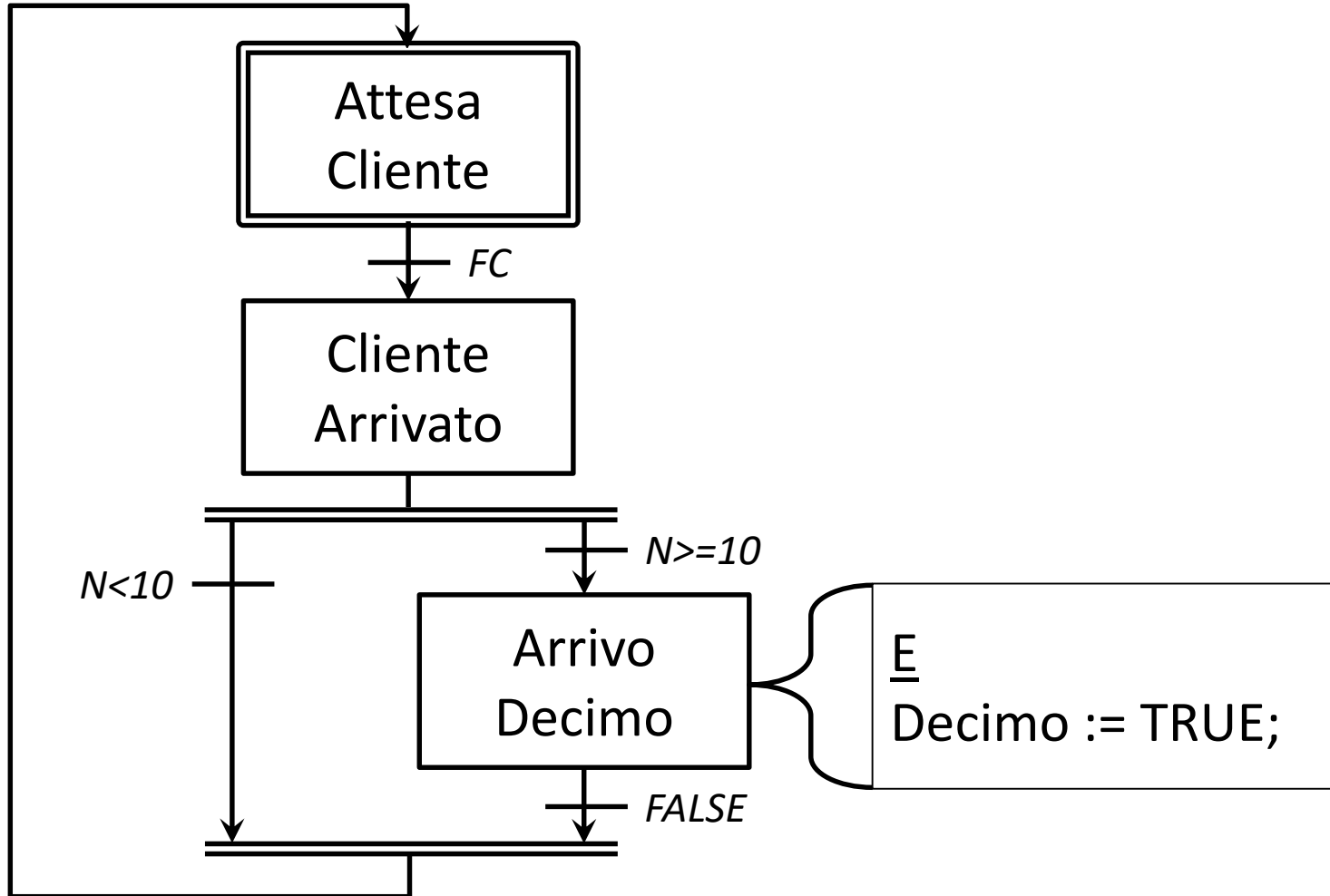
Esempio azioni in altri linguaggi



Abbiamo la possibilità di aggiungere azioni sull'ingresso (E), cicliche (C) e all'uscita (X), in questo caso utilizziamo il testo strutturato

Esempio

Esempio azioni in altri linguaggi



Da SFC a Ladder

Esiste la possibilità di tradurre un programma implementato in SFC in un programma Ladder, attraverso semplici regole, che prescindono dal programma stesso.

Nel corso ci concentriamo sull'algoritmo senza ricerca di stabilità.

N.B.: sono regole standard, applicabili a tutti i programmi SFC!

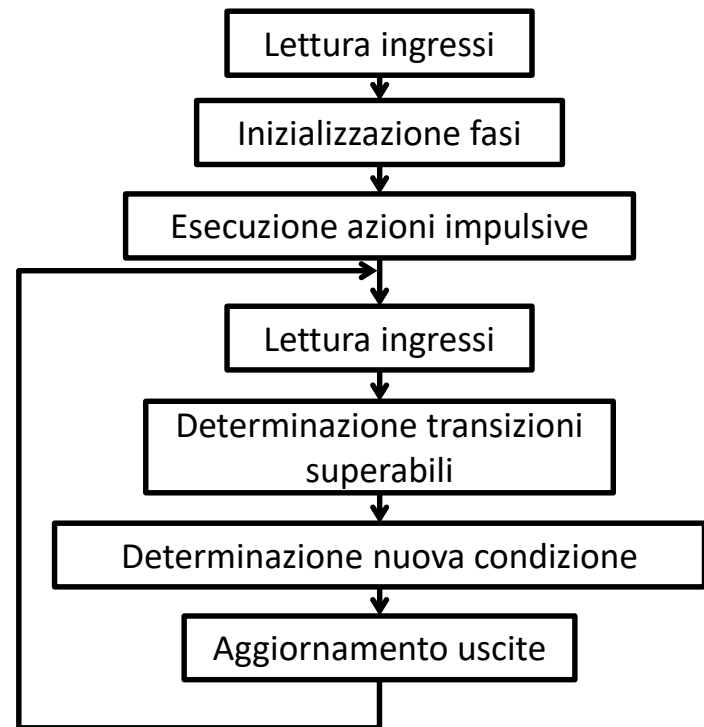
Da SFC a Ladder

Consideriamo questo algoritmo di evoluzione del SFC

Questo algoritmo è detto senza ricerca di stabilità.

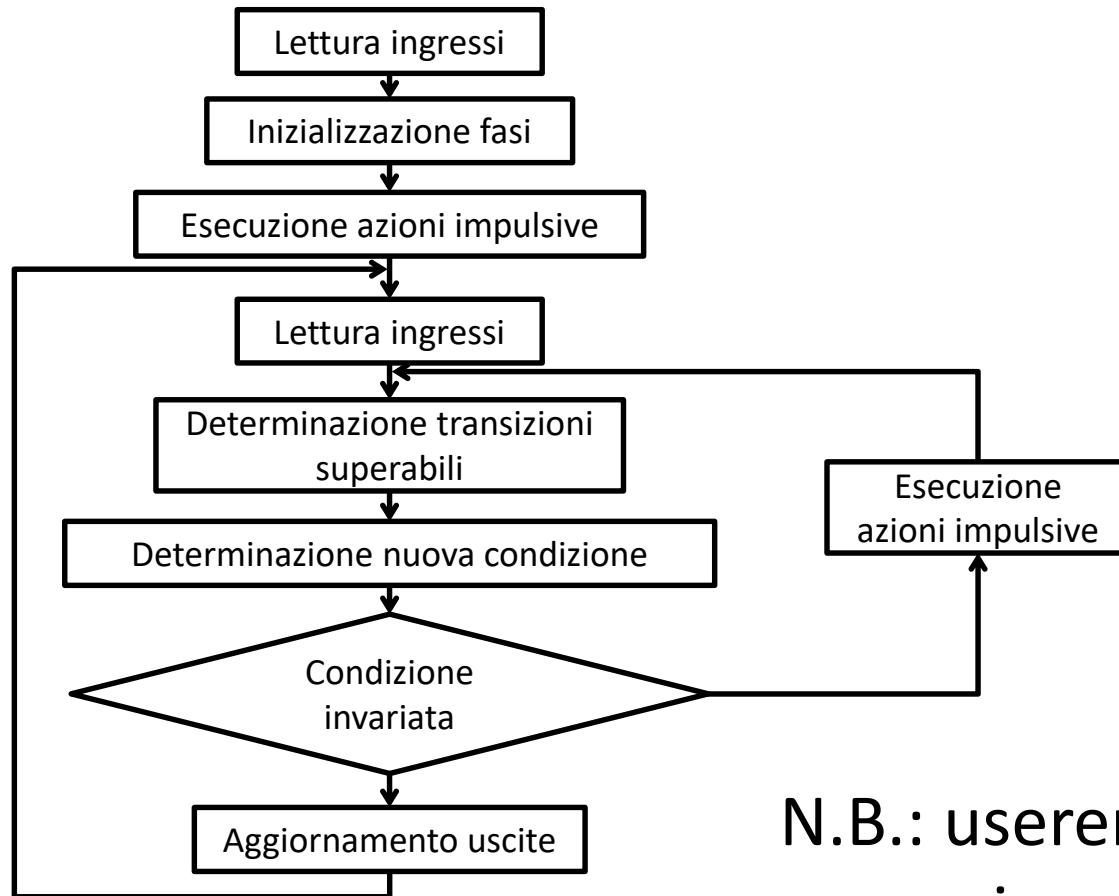
Nel caso di fase non stabile (fase con, all'attivazione, la transizione di uscita già vera), questo schema «commette un errore»:

- Se l'azione fosse continua non dovrebbe essere attivata
- Se l'azione fosse impulsiva si



Da SFC a Ladder

A titolo informativo, ecco l'algoritmo con ricerca di stabilità



N.B.: useremo l'algoritmo senza ricerca di stabilità

Da SFC a Ladder

Il ladder deve essere costituito da quattro sezioni:

- Inizializzazione

Eseguita solo durante il primo step del programma

- Esecuzione delle azioni

Abilitazione delle azioni connesse alla fase attiva

- Valutazione delle transizioni

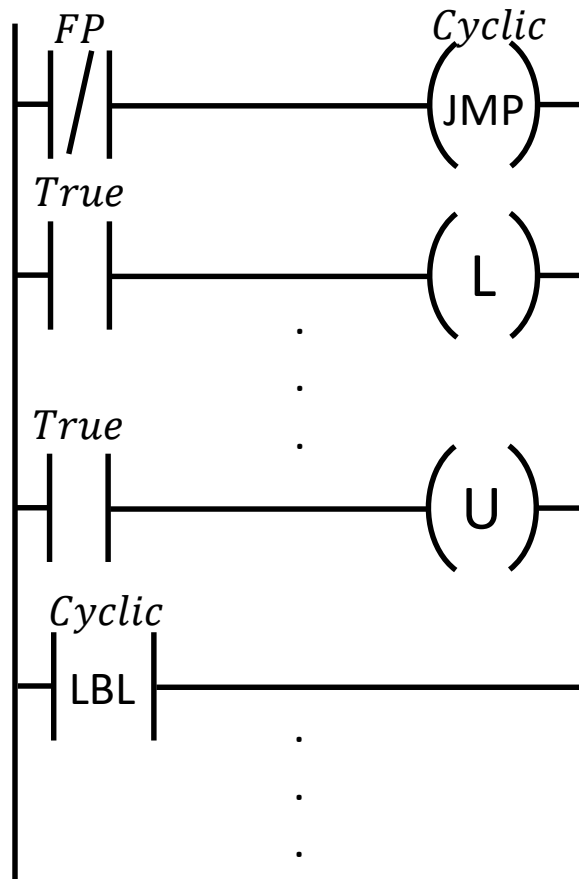
Abilitazione delle transizioni connesse alle fasi attive

- Aggiornamento condizione

Verifica del passaggio tra le fasi attuali e le seguenti

Da SFC a Ladder

Inizializzazione

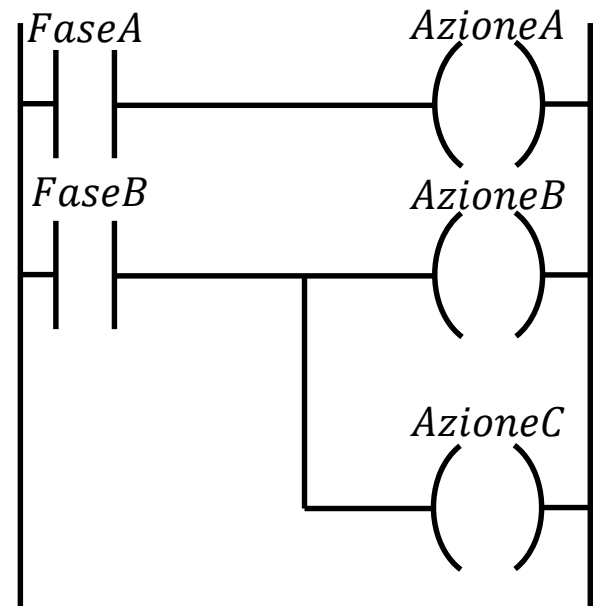


N.B.: questa parte del software, in AS, può essere disposta nella vera e propria `_INIT!`

Da SFC a Ladder

Esecuzione delle azioni

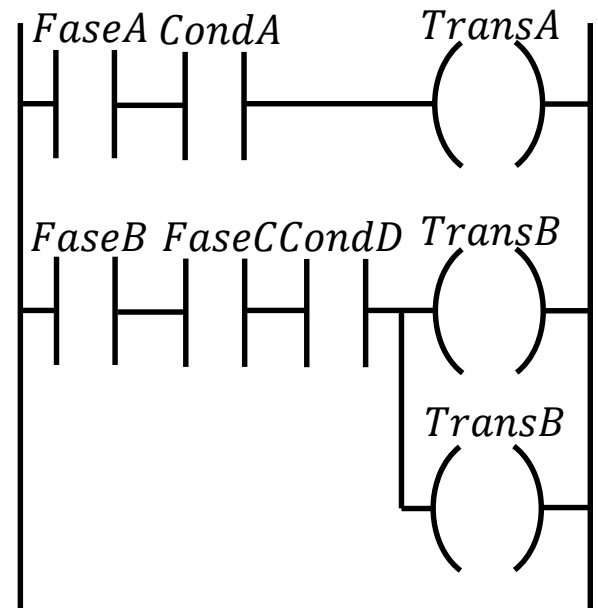
- Per ogni azione continua si usa un rung in cui, in OR, vengono messe tutte le fasi che abilitano quell'azione
- Per le azioni memorizzate si usano bobine SET – RESET
- Per le azioni impulsive serve costruire la rete che genera l'impulso



Da SFC a Ladder

Valutazione delle condizioni

- Per ogni fase, si verificano le condizioni di uscita che abilitano la transizione
- Nel caso di transizioni connesse a più di una fase (parallelo) si devono mettere in AND i segnali booleani delle fasi



Da SFC a Ladder

Aggiornamento condizione

In corrispondenza di una transizione attiva, vengono disabilitate tutte le fasi a monte e abilitate tutte quelle a valle

