Corso di
Sistemi e Reti 1

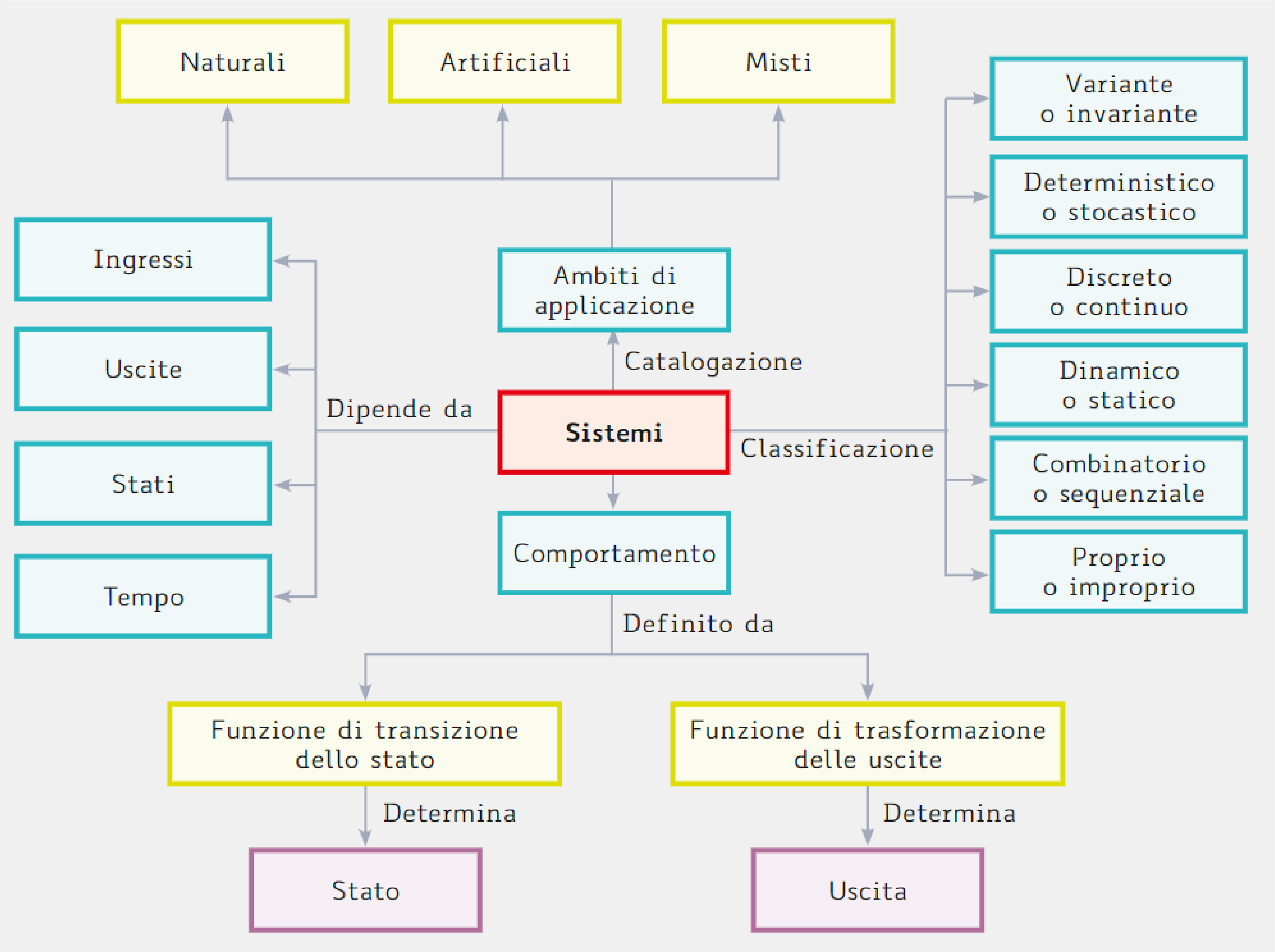
Teoria

- I sistemi e gli automi
- Le architetture dei sistemi di elaborazione
- La programmazione Assembly
- Fondamenti di networking
- Le reti Ethernet e lo strato di collegamento

Laboratorio

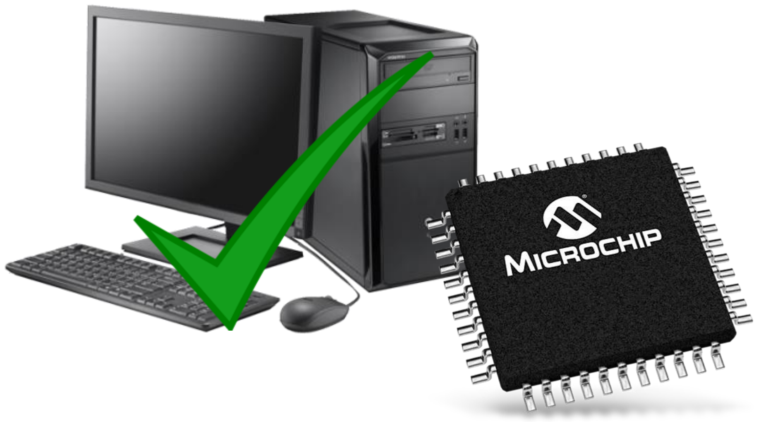
- Simulazione degli automi con JFLAP
- Assemblaggio di un calcolatore
- Programmazione Assembly con Turbo assembler
- Esercitazioni con Arduino e Raspberry
- Comunicazione tra Arduino e PC
- Cablaggio e realizzazione cavi di rete
- Utilizzo di Wireshark
- Il protocollo Ethernet e ARP
- Introduzione all’uso di CISCO Packet Tracer
- Progettazione di siti web statici (HTML, CSS, Framework)
- Linguaggio XML

Conosciamo i sistemi


Naturali
Artificiali
Misti
Ingressi
Uscite
Stati
Tempo
Ambiti di applicazione
Catalogazione
Dipende da
Sistemi
Classificazione
Variante o invariante
Deterministico o stocastico
Discreto o continuo
Dinamico o statico
Combinatorio o sequenziale
Proprio o improprio
Comportamento
Definito da
Funzione di transizione dello stato
Funzione di trasformazione delle uscite
Determina
Determina
Stato
Uscita

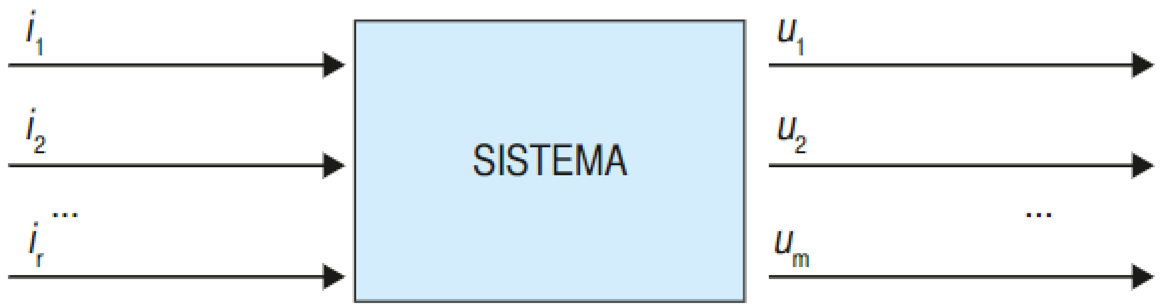
Sistema

Un sistema è un insieme di elementi in relazione tra di loro secondo leggi ben precise che concorrono al raggiungimento di un obiettivo comune.

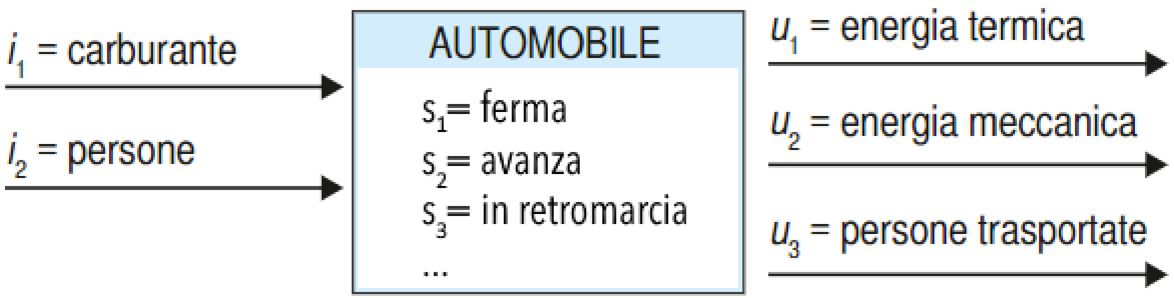


È un oggetto che reagisce o si evolve mediante leggi proprie ed è formato da elementi diversi, interconnessi tra di loro e con l'ambiente esterno.

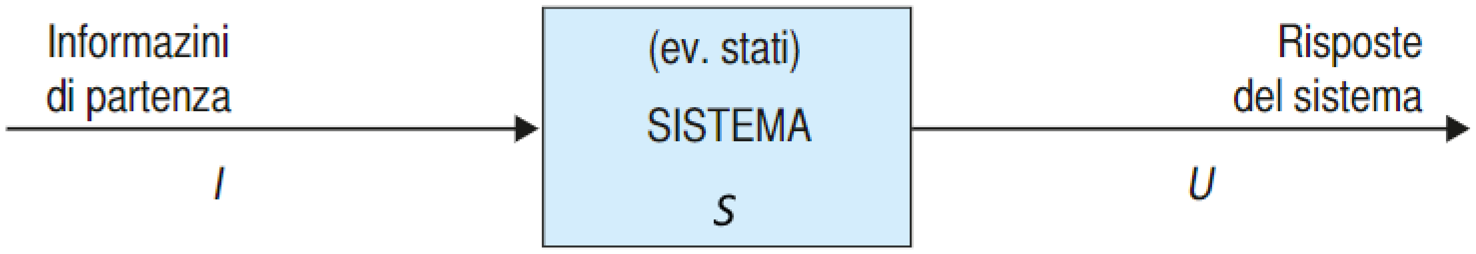
Modello di sistema



Ad esempio per un sistema Automobile:



COMPORTAMENTO → MODELLO



Ingresso (azione, causa)

Uscita (reazione, effetto)

Problematiche di interesse nello studio dei sistemi:

- **Previsione** noti S, I ⟶ trovare U
- **Controllo** noti S, U_{des} ⟶ trovare I
- **Identificazione** noti U, I ⟶ trovare S

Le 4 grandezze che determinano univocamente un sistema descrivendone il comportamento:

INGRESSI		
- Le variabili d'ingresso I - I valori degli ingressi VI	$I = i_1, i_2, \dots i_n$ con n variabili d'ingresso $VI = (vi_{1,1}, vi_{2,1}, \dots vi_{n,1}) \dots (vi_{1,m}, vi_{2,m}, vi_{n,m})$	**m** gruppi di **n** valori di ingresso

USCITE		
- Le variabili d'uscita U - I valori delle uscite VU	$U = u_1, u_2, \dots u_n$ con r variabili d'ingresso $VU = (vu_{1,1}, vu_{2,1}, \dots vu_{r,1}) \dots (vu_{1,s}, vu_{2,s}, vu_{r,s})$	**s** gruppi di **r** valori d'uscita

STATO		
Stato del sistema interno (*S*) in un singolo istante ***t***	$S = s_1, s_2, \dots s_t$	**t** stati in cui può trovarsi il sistema

TEMPO		
Insieme ***T*** dei tempi in cui viene studiato il sistema	$T = t_1, t_2, \dots t_q$	**q** istanti in cui si studia il sistema

Stato

Insieme di informazioni necessarie e sufficienti per descrivere le condizioni in cui si trova il sistema in un qualunque istante.
Il sistema, in base ai dati ricevuti in ingresso ed alla sua natura, evolve istante dopo istante in una nuova situazione che è chiamata **stato**.
Possiamo paragonare lo stato ad una fotografia del sistema in un determinato istante. Leggere i valori dello stato di un sistema in un determinato momento equivale a farne una fotografia istantanea: i valori degli ingressi, delle uscite e quelli assunti dallo stato interno di un sistema, vengono memorizzati in apposite variabili che prendono il nome di variabili d'**ingresso**, d'**uscita** e di **stato**.

Esempio1:
In un sistema Contatore di elettricità possiamo paragonare lo stato al conteggio di kW consumati fino a quell'istante

Esempio2:
In un sistema Automobile, lo stato può essere rappresentato dalla velocità attuale dell'automobile stessa (che può essere 0 se è ferma)

Esempio3:
In un sistema Aeroplano, lo stato può essere rappresentato dall'insieme dei valori che rappresentano i litri di carburante nei serbatoi, il numero di giri del motore, il livello dell'olio etc.

Funzione di <u>transizione</u> dello stato "*f*" (evoluzione temporale dello stato)

$\boxed{s(t) = f\,(s(t_0),\, i(t))}$ o in modo più formale: $s(t) = f\,(t,\, t_0,\, s(t_0),\, i(\bullet))$

È la funzione che determina il valore dello stato del sistema **s**(t) in un generico istante **t**, in base:

- alla situazione iniziale **s**(t_0) (valore iniziale dello stato del sistema) e
- a tutti gli ingressi applicati al sistema all'istante iniziale t_0 (con $t_0 < t$) fino all'istante t, cioè **i**(•) funzione d'ingresso definita nell'intervallo [t_0, t]

Funzione di <u>trasformazione</u> delle uscite "*g*" (evoluzione temporale dell'uscita)

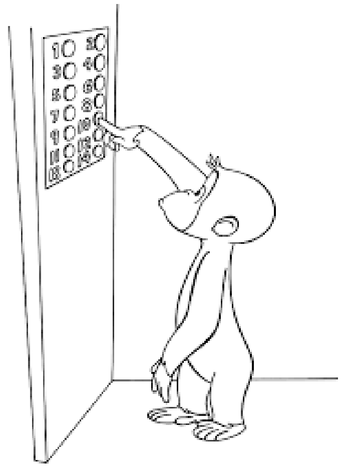
$\boxed{u(t) = g\,(s(t),\, i(t))}$ o in modo più formale: $u(t) = g\,(t,\, s(t),\, i(t))$

È la funzione che determina il valore che avrà l'uscita **u**(t) a un generico istante **t** (evoluzione temporale dell'uscita), conoscendo il valore dello stato (**s**(t)) e degli ingressi (**i**(t)) nel medesimo istante

Esempio1:

Un ascensore che serve una casa di due piani può essere rappresentato come un sistema nel modo che segue:

STATI S: {PT,P1, P2}	**INGRESSI** I: {T,1,2}	**USCITE** U: {su, giù, fermo}
PT: Piano terra P1: 1° piano P2: 2° piano	T: Pulsante per piano terra 1: Pulsante per 1° piano 2: Pulsante per 2° piano	azione compie l'ascensore in funzione di ingressi e stati

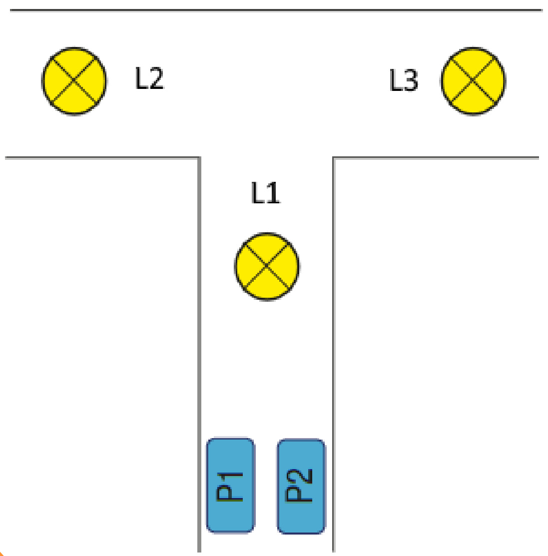


Funzione di transizione dello stato "*f*"	Funzione di trasformazione delle uscite "*g*"
f (PT, T) = PT f (PT, 1) = P1 f (PT, 2) = P2 f (P1, T) = PT f (P1, 1) = P1 f (P1, 2) = P2 f (P2, T) = PT f (P2, 1) = P1 f (P2, 2) = P2	g (PT, T) = fermo g (PT, 1) = su g (PT, 2) = su g (P1, T) = giù g (P1, 1) = fermo g (P1, 2) = su g (P2, T) = giù g (P2, 1) = giù g (P2, 2) = fermo

<u>*Esempio2:*</u>

Prendiamo il sistema di illuminazione rappresentato a destra, ovvero un edificio con un ingresso e due corridoi, uno sinistro e uno destro.
Un esempio di comportamento del sistema può essere descritto con le seguenti tabelle:

STATI (T0)	P1 OFF P2 OFF	P1 ON P2 OFF	P1 OFF P2 ON	P1 ON P2 ON
Buio	Buio	Luce ingresso	Luce ingresso	Non abilitato
Luce ingresso	Luce ingresso	Luce sinistra	Luce destra	Non abilitato
Luce sinistra	Luce sinistra	Buio	Buio	Non abilitato
Luce destra	Luce destra	Buio	Buio	Non abilitato



*nella tabella vengono determinati gli stati sulla base degli ingressi e degli stati precedenti, pertanto potremmo parlare di funzione di transizione: **f***

N.B.
Ho definito io che ci fossero questi stati $s(t_1)$ in base agli stati $s(t_0)$ e ai valori degli ingressi P1 e P2.
È un esempio.

STATI (T1)	USCITE ← (T1)		
Buio	L1 = Off	L2 = Off	L3 = Off
Luce ingresso	L1 = On	L2 = Off	L3 = Off
Luce sinistra	L1 = Off	L2 = On	L3 = Off
Luce destra	L1 = Off	L2 = Off	L3 = On

*nella tabella vengono determinate le uscite sulla base degli stati nello stesso istante, pertanto potremmo parlare di funzione di trasformazione: **g***

In questa tabella possiamo notare come, variando lo stato (scorrendo la prima colonna dall'alto al basso per esempio), vengano modificate istantaneamente le uscite. Infatti al variare dello stato, per esempio Buio, otteniamo lo spegnimento delle 3 lampadine.

<u>Esempio3:</u>

Una caffettiera (moka) è formata dai seguenti elementi:

- *recipiente superiore per il caffè*
- *serbatoio per l’acqua*
- *filtro*
- *guarnizione di tenuta vapore*
- *manico protettivo*

Individuiamo:

STATI S: {S1, S2, S3, S4, S5, S6}	**INGRESSI** I: {I1, I2, I3}	**USCITE** U: {U1, U2}
S1: Vuota S2: Pronta all’uso S3: In riscaldamento S4: Ebollizione S5: Caffè da versare S6: Da ripulire	I1: Acqua I2: Caffè I3: Energia termica	U1: Caffè U2: Vapore

STATI (T0)	I1 ACQUA I2 NO I3 NO	I1 ACQUA I2 CAFFÈ I3 NO	I1 ACQUA I2 CAFFÈ I3 ENERGIA TERMICA	I1 ACQUA I2 NO I3 ENERGIA TERMICA	I1 NO I2 CAFFÈ I3 ENERGIA TERMICA	I1 NO I2 NO I3 ENERGIA TERMICA
Vuota	Vuota	Pronta all'uso	Pronta all'uso	In riscaldamento	Non abilit.	Non abilit.
Pronta all'uso	Pronta all'uso	Pronta all'uso	In riscaldamento	Non abilit.	Non abilit.	Non abilit.
In riscaldamento	Pronta all'uso	Pronta all'uso	In ebollizione	Non abilit.	Non abilit.	Non abilit.
In ebollizione	Pronta all'uso	Pronta all'uso	Caffè da versare	Non abilit.	Non abilit.	Non abilit.
Caffè da versare	Da ripulire	Non abilit.	Da ripulire	Non abilit.	Non abilit.	Non abilit.
Da ripulire	Da ripulire	Da ripulire	Non abilit.	Non abilit.	Non abilit.	Non abilit.

*tabella di transizione dello stato (rappresenta la funzione **f**)*

STATI (T1)	USCITE ← (T1)	
Vuota	U1: Nulla	U2: Nulla
Pronta all'uso	U1: Nulla	U2: Nulla
In riscaldamento	U1: Nulla	U2: Nulla
In ebollizione	U1: Nulla	U2: Vapore
Caffè da versare	U1: Caffè	U2: Vapore
Da ripulire	U1: Nulla	U2: Nulla

*tabella di trasformazione delle uscite (rappresenta la funzione **g**)*

Esercizi

1) Individua gli ingressi, le uscite e gli stati che caratterizzano i seguenti sistemi:
   a) sistema emettitore di biglietti (prezzo 50 centesimi, 1 euro, accetta monete da 50, 20, 10 centesimi)
   b) sistema termostato di casa per la regolazione della temperatura di un’abitazione
   c) sistema serbatoio idrico che si riempie d’acqua automaticamente quando il volume dell’acqua all’interno scende al di sotto di un certo livello (aggiungi le ipotesi mancanti)
   d) sistema distributore automatico di lattine per l’erogazione di due tipi di lattine (inserendo tre tipi di monete)
   e) sistema distributore di benzina per effettuare il pieno all’automobile

2) Dato un sistema moto scooter, individua gli insiemi degli ingressi, uscite e stati che lo caratterizzano in base ai seguenti diversi obiettivi di studio:
   a) stabilire l’andamento del moto del veicolo, cioè restituire come uscita le coordinate della posizione, la velocità e l’accelerazione istantanee, al variare del tempo
   b) stabilire il livello di usura degli organi meccanici in funzione della frequenza dei cambi d’olio
   c) stabilire l’autonomia massima di strada percorribile con un unico pieno di carburante

Catalogazione dei sistemi

1) Per ambiti di applicazione:
   - sistemi naturali *(es. il sistema circolatorio dell'uomo, un albero da frutto etc.)*
   - sistemi artificiali *(es. un distributore di bibite o un ascensore)*
   - sistemi misti *(es. una centrale telefonica)*

2) Per caratteristiche:
   - discreto o continuo
   - variante o invariante
   - statico o dinamico
   - deterministico o stocastico
   - combinatorio o sequenziale
   - lineare

Sistema discreto

<u>Discreto nel tempo</u> (nell'avanzamento):
quando è discreto l'insieme dei tempi T in cui viene studiato ovvero non esistono tempi intermedi di osservazione tra un istante di osservazione t_n e il suo successivo t_{n+1}

es. ripresa cinematografica in cui ci sono 40ms tra un fotogramma e il successivo (ci sono 25fps) o microprocessore scandito dal clock (orologio di sistema)

<u>Discreto nelle sollecitazioni</u>:
quando l'insieme delle variabili ingresso (VI) è discreto (una variabile è discreta quando può assumere solo un numero finito di valori)

es. sistema di illuminazione che si accende con pulsanti o distributore di lattine che compro con monete

<u>Discreto nelle interazioni</u>:
quando la funzione di transizione (e quindi lo stato futuro) e/o la funzione di trasformazione (e quindi le uscite) sono entrambe discrete

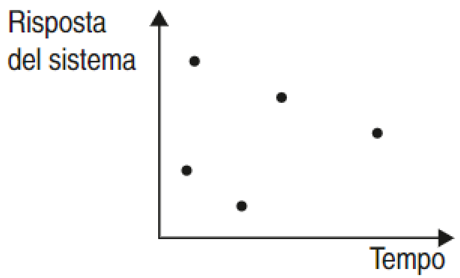
es. orologio digitale che, alimentato con tensione continua, mostra le cifre in modo discreto, cambiando ad ogni sec

Sistema continuo

Sistema continuo (o non discreto) contemporaneamente:
- nel tempo (avanzamento)
- nelle sollecitazioni (input)
- nelle interazioni

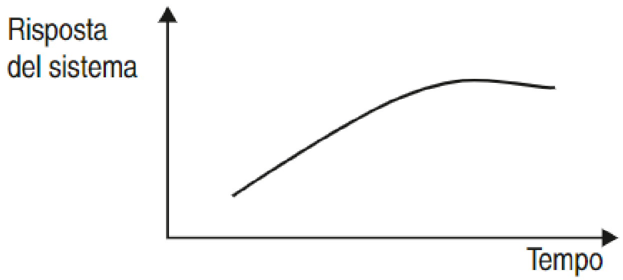
es. rubinetto dell'acqua

Sistema tempo-discreto



$T \subseteq \mathbb{Z}$

Sistema tempo-continuo



$T \subseteq \mathbb{R}$

Sistema variante nel tempo

Quando nel tempo varia la sua funzione di transizione di stato (es. lo stato futuro è in deterioramento o miglioramento)

es. un missile in movimento la cui massa si riduce per il consumo di combustibile o
il cilindro di un motore che viene deteriorato dallo sfregamento del pistone

Sistema invariante nel tempo (stazionario)

Quando i parametri che lo caratterizzano rimangono invarianti nel tempo e di conseguenza anche le leggi che legano le sollecitazioni alle risposte rimangono invariate nel tempo. L'idea è che l'esito di ogni esperimento non dipende dall'istante in cui si effettua

$$f\,(t, t_0, s(t_0), i(\bullet)) = f\,(t+ \Delta t_0, t_0 + \Delta t_0, s(t_0), i^{\Delta t0}(\bullet))$$

es. le leggi della fisica sono tempo-invarianti (per quanto se ne sa allo stato attuale)

Non bisogna confondersi e considerare variante un sistema quando variano gli ingressi e/o le uscite: in tal caso il sistema è semplicemente da definirsi dinamico

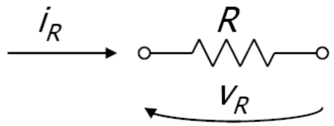
Sistema statico

Quando il legame ingresso-uscita è istantaneo o statico

$$u(\mathrm{t}) = g\ (i(\mathrm{t}))\ ,\ \forall \mathrm{t}$$

cioè il valore dell'uscita *u* all'istante *t* dipende solo dal valore dell'ingresso *i* allo stesso istante *t* (non dai valori precedenti)

es.1 Resistore ideale



$$i(t) = i_R(t)$$
$$u(t) = v_R(t) = R\, i_R(t) = g\big(i(t)\big),\ \forall t$$

es.2 Il sistema elettrico composto da una lampadina, se consideriamo che la relazione ingresso- uscita è istantanea e non varia nel tempo: o è accesa o è spenta ed il passaggio avviene istantaneamente.

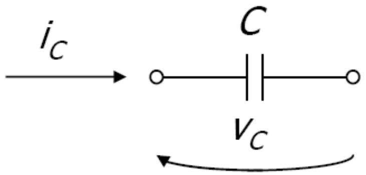
Sistema dinamico

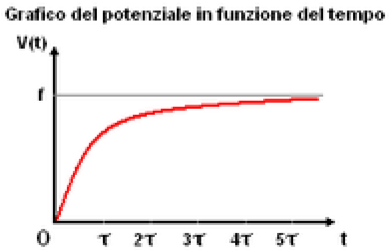
Quando il legame ingresso-uscita è di tipo dinamico

$$u(\mathrm{t}) = g\ (i(]-\infty,\mathrm{t}]))\ ,\ \forall \mathrm{t}$$

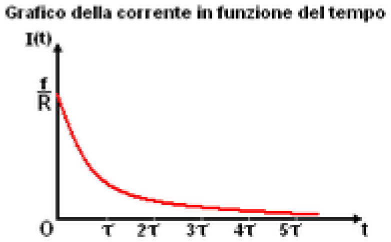
cioè il valore dell'uscita *u* all'istante *t* dipende da tutti i valori assunti dall'ingresso *i* fino all'istante *t*.

es.1 Condensatore ideale





$$i(t) = i_C(t) = C\, dv_C(t)/dt$$
$$u(t) = v_C(t) = \frac{1}{C}\int_{-\infty}^{t} i_C(\sigma)d\sigma =$$
$$= \frac{1}{C}\int_{-\infty}^{t} i(\sigma)d\sigma = g\big(i(]-\infty,t])\big),\ \forall t$$



Per riassumere tutta la "**storia passata**" del sistema fino all'istante t_0, si può introdurre lo **stato** $\mathbf{s(t_0)}$ che racchiude in sé tutta la memoria del passato.
Un **sistema** è allora **dinamico se**:

$$u(t) = g\,(s(t_0), i([t_0,t]))\ ,\ \forall t \geq t_0$$

Allora, riprendendo l'esercizio del condensatore ideale:

$$\boldsymbol{s(t_0)} = \frac{1}{C}\int_{-\infty}^{t_0} i_C(\sigma)d\sigma = \boldsymbol{v_C(t_0)}$$

$$u(t) = \frac{1}{C}\int_{-\infty}^{t} i_C(\sigma)d\sigma = \underbrace{\frac{1}{C}\int_{-\infty}^{t_0} i_C(\sigma)d\sigma}_{s(t_0)} + \frac{1}{C}\int_{t_0}^{t} i_C(\sigma)d\sigma =$$

$$= s(t_0) + \frac{1}{C}\int_{t_0}^{t} u(\sigma)d\sigma = g\left(s(t_0), u([t_0,t])\right), \quad \forall t \geq t_0$$

Sistema deterministico

Quando le funzioni di transizione e di trasformazione permettono la determinazione del valore dello stato e delle uscite in modo univoco.

Sistema stocastico

Quando almeno una delle due funzioni è regolata da legami di natura probabilistica.

es.1 Il numero che viene estratto nel gioco della Roulette è assolutamente causale, possiamo pertanto affermare che il sistema è di tipo stocastico

es.2 La luminosità di una stanza alle ore 10 di ogni giorno è un sistema stocastico, perché la quantità di luce dipende dalle condizioni atmosferiche, che sono di tipo probabilistico

es.3 Un albero da frutta è un sistema stocastico in quanto la quantità di frutta prodotta annualmente non viene determinata analiticamente

es.4 La distanza tra elettrone e nucleo dell’atomo di Idrogeno si ottiene con procedimento stocastico.

Sistema combinatorio

Quando le uscite dipendono solo dai valori presenti agli ingressi nel medesimo istante ovvero sono una combinazione degli ingressi nello stesso istante.
Si dice che le uscite non ricordano il passato e che il sistema è senza memoria.

es.1 Serratura di una cassaforte. Nel sistema Cassaforte per ottenere la combinazione si devono ruotare e posizionare le tre manopole presenti ciascuna su di un numero: è indipendente dal valore iniziale delle manopole in quanto conta solamente il numero finale sul quale vengono posizionate.

es.2 Porte logiche (AND, OR, NOT,…). Un circuito digitale costituito da sole porte logiche è combinatorio, così come lo sono le singole porte logiche: l'uscita dipende unicamente dal valore presente sugli ingressi e non dai valori degli ingressi delle operazioni precedenti.

Sistema sequenziale

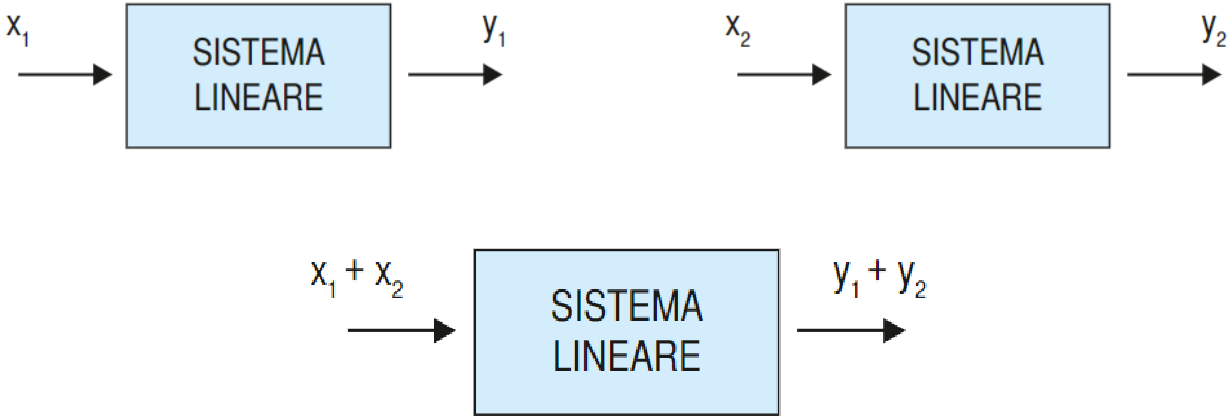
Quando le uscite dipendono non solo dai valori degli ingressi in quell'istante, ma anche dai valori assunti in precedenza dagli ingressi.

es. Un distributore automatico di merendine dove l'uscita o meno del prodotto, dipende dalle monete inserite precedentemente, che concorrono a formare l'importo necessario

Sistema lineare

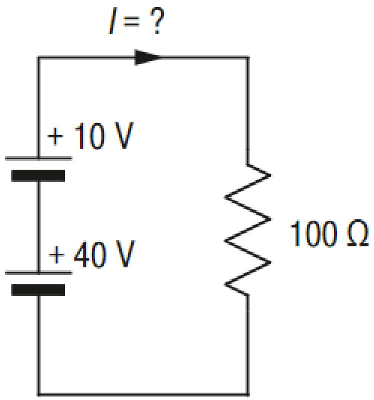
Quando è possibile applicare la sovrapposizione degli effetti, ovvero vale il **principio di sovrapposizione degli effetti** il quale prevede che gli effetti dovuti a più cause sono la risultante (somma algebrica) degli effetti prodotti da ogni causa applicata singolarmente.

Avendo quindi, ad esempio, due cause x_1 e x_2 che generano un input, in un sistema lineare, è possibile farle agire singolarmente e ricavare il risultato parziale delle uscite y_1 e y_2 e quindi effettuarne la somma algebrica per ottenere l'effetto totale



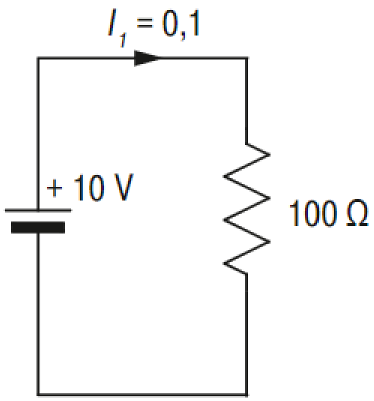
Un sistema lineare si ha dove l'espressione matematica che governa la relazione ingresso-uscita è rappresentata da un'equazione lineare, ovvero contiene solo operazioni di tipo lineare, come ad esempio un prodotto per una costante.
Un esempio è allora la legge di Ohm che mette in relazione lineare corrente (ingresso) e tensione (uscita) in una resistenza.
Questa relazione è chiara se si traccia un grafico ingresso-uscita che, nel caso di sistema lineare, dovrebbe mostrare una retta.

es. Applichiamo il principio di sovrapposizione degli effetti in un semplice circuito lineare elettrico contenente due generatori di tensione, rispettivamente da 10 V e 40 V e una resistenza da 100 Ω.

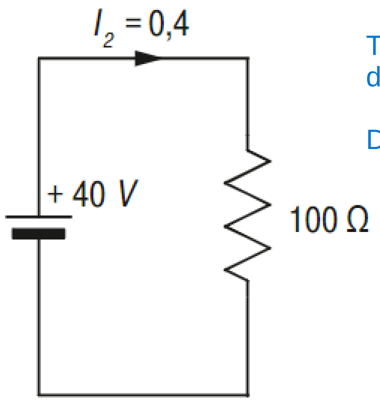


Per calcolare la corrente totale circolante nel resistore, facciamo agire un generatore per volta, calcolando le correnti parziali. Infine sommando gli effetti parziali otteniamo il risultato atteso, cioè la corrente I totale circolante:

$I_{TOT} = I_1 + I_2 = 0{,}1 + 0{,}4 = 0{,}5$ A



solo il primo generatore



solo il secondo generatore

Tabella degli stati: tutte le combo degli input che danno degli output

Discreto = puoi misurarlo

A	B
0	0
0	1
1	0
1	1

Conosciamo i sistemi

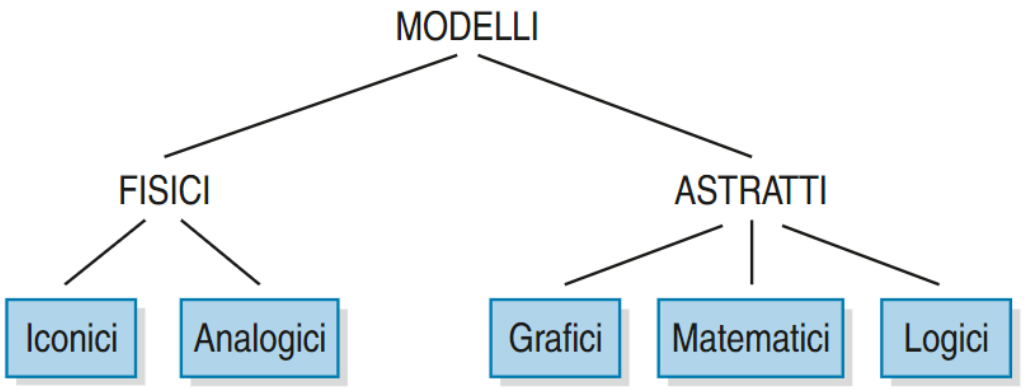
Sistema reale

I sistemi reali sono generalmente complessi da studiare, e sono spesso formati da elementi molto numerosi che possono avere delle relazioni tra di loro non facilmente intuitive.
Per studiare un sistema è necessario utilizzare un modello che ne simuli il comportamento focalizzando l'attenzione sugli elementi essenziali, trascurando i dettagli che non portano informazioni e rendono lo studio eccessivamente complesso.

Modellizzazione

Modello

Un modello di un sistema è quell'insieme di elementi che ci permettono di riprodurre, valutare o stimare, anche se <u>limitatamente ad un certo contesto</u>, le funzioni svolte dal sistema originale, in maniera più semplice senza dover necessariamente intervenire su di esso.

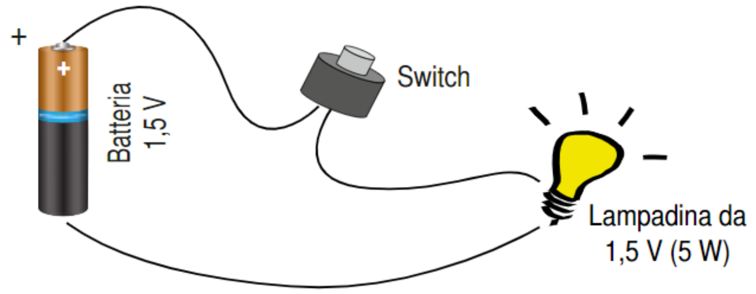


es. **MODELLO SISTEMA BATTERIA-PULSANTE-LAMPADINA**

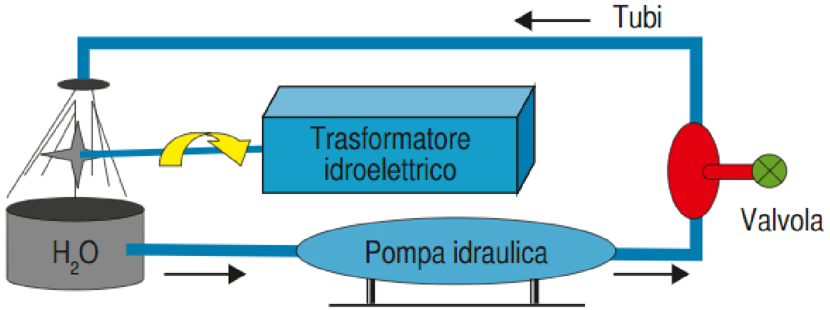
In questo esempio vogliamo studiare un sistema formato dai seguenti componenti elettrici:

- una batteria da 24 V (5A);
- un pulsante a relè;
- due cavi di collegamento;
- una lampadina da 100 W.

Iniziamo dando una **rappresentazione iconica** al sistema:
utilizziamo un modello in scala ridotta, con una batteria stilo da 1,5 V, un micro switched ed una mini lampadina da 5 W



Un **modello analogico** può essere quello in cui il flusso di corrente elettrica viene sostituito dal fluire dell'acqua all'interno dei tubi, la batteria da una pompa idraulica e il pulsante da una valvola in grado di aprire e chiudere il flusso. La funzione della lampadina è simulata tramite un trasformatore idroelettrico in grado di convertire l'energia di pressione dell'acqua in energia elettrica



La rappresentazione mediante **modello matematico** prevede innanzitutto la definizione degli ingressi, delle uscite e degli stati, e quindi delle funzioni di transizione e trasformazione.

Ingresso: condizione del pulsante di comando:
I = {I(t)}
Variazione dell'ingresso: 0 per pulsante a riposo / 1 per pulsante premuto
VI = {0,1}

Uscita: condizione della lampadina
U = {U(t)}
Variazione dell'uscita: 0 per lampadina spenta / 1 per lampadina accesa
VU = {0,1}

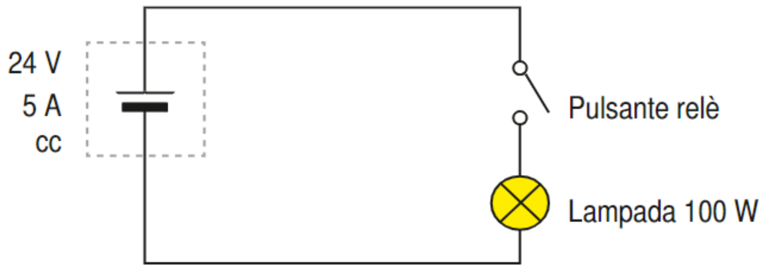
Stato: passaggio di corrente nel circuito
S = {S(t)}
0 indica che non passa corrente/ 1 indica che passa corrente

Funzione di transizione f():
lo stato successivo dipende dall'ingresso precedente I(t)

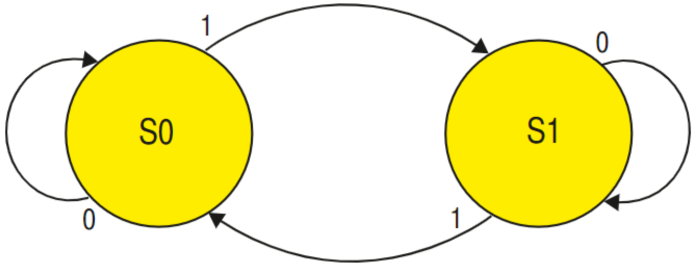
$$S(t+1) = \begin{cases} \bullet \quad S(t) \text{ quando } I(t) = 0 \\ \bullet \quad 1 - S(t) \text{ quando } I(t) = 1 \quad \textit{(stato complementare)} \end{cases}$$

Funzione di trasformazione g():
l'uscita dipende dallo stato nello stesso istante
U(t) = g(t, S(t)) = S(t)

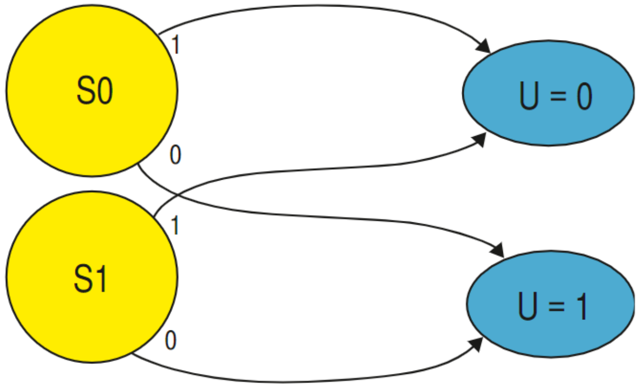
La rappresentazione del sistema mediante **modello grafico** prevede che vengano indicati sotto forma di circuiti elettrici i diversi componenti e le diverse grandezze fisiche in gioco; in questo caso abbiamo un pulsante a relè, una lampada da 100 W e un alimentatore da 24 Ve 5A a corrente continua.



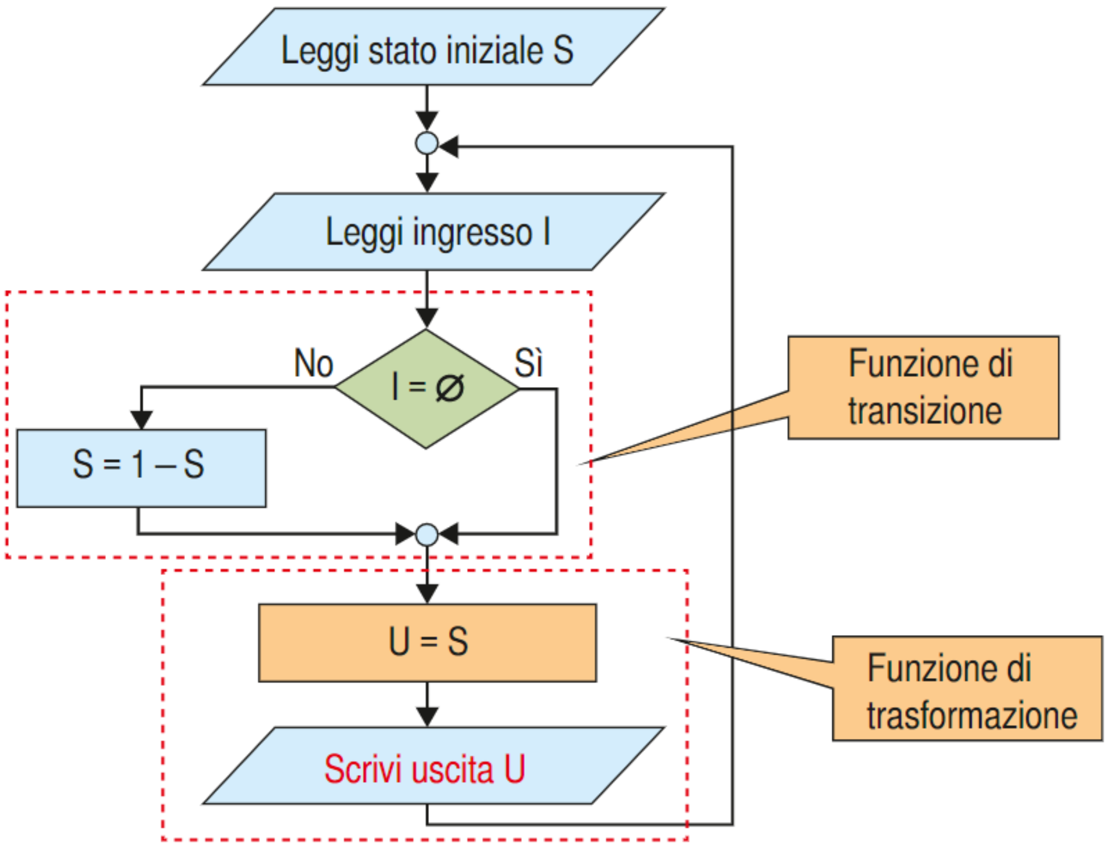
La rappresentazione del sistema mediante modello grafico può essere anche espressa attraverso il **grafo degli stati** (chiamato anche **grafo delle transizioni**), attraverso il quale devono essere disegnati tutti i possibili stati con dei cerchi (nel nostro caso 0 e 1), quindi le linee che rappresentano gli ingressi applicati, collegati allo stato successivo al quale viene portato il sistema.

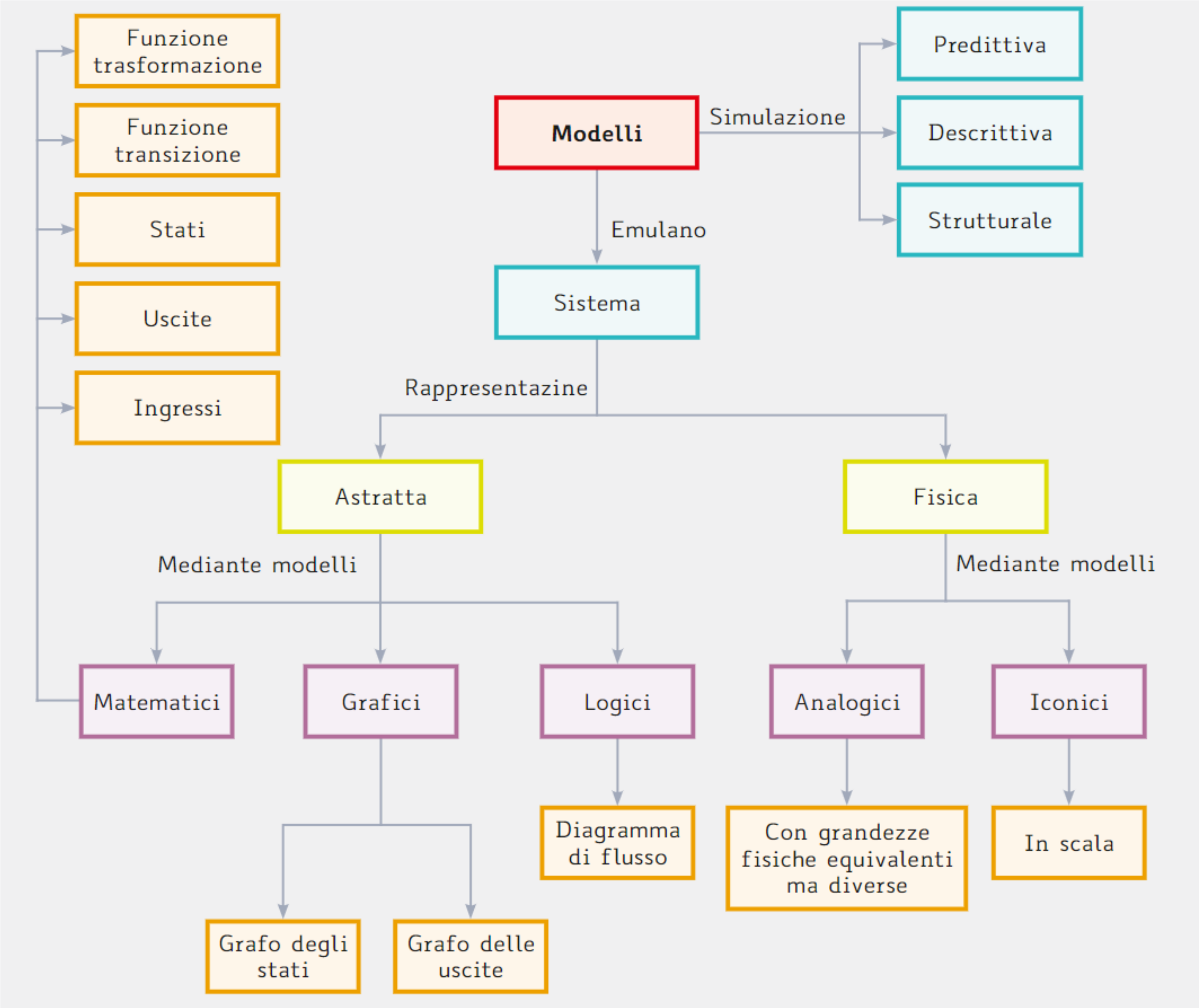


La rappresentazione del sistema mediante **grafo delle uscite** o **grafo delle trasformazioni** prevede che per prima cosa si disegnino tanti cerchi quanti sono i possibili stati, quindi si traccino poi le linee che rappresentano gli ingressi applicati, collegandole alle uscite a cui viene portato il sistema. Infine tracciamo tanti ellissi quante sono le uscite.



La rappresentazione del sistema mediante **modello logico** prevede un diagramma di flusso che di volta in volta valuta i valori dell'uscita e dello stato in funzione dell'ingresso.




Funzione trasformazione
Funzione transizione
Stati
Uscite
Ingressi
Modelli
Simulazione
Predittiva
Descrittiva
Strutturale
Emulano
Sistema
Rappresentazine
Astratta
Fisica
Mediante modelli
Mediante modelli
Matematici
Grafici
Logici
Analogici
Iconici
Diagramma di flusso
Con grandezze fisiche equivalenti ma diverse
In scala
Grafo degli stati
Grafo delle uscite

Gli automi a stati finiti

- Definizioni
- Proprietà degli automi
- Esempi di studio di automi
- Automa riconoscitore
- Mappa concettuale

Automa a stati finiti

È una macchina automatizzata che ripete in maniera automatica le operazioni svolte, sostituendo l'uomo in compiti ripetitivi, faticosi e pericolosi.
È un sottoinsieme **dinamico** di un sistema: un sistema **invariante**, **deterministico**, **discreto** negli avanzamenti e nelle interazioni, in cui gli insiemi VI e VU, così come gli stati, sono composti da un numero finito di elementi.

Esempi: *una lavatrice, un distributore di bevande automatico, un ascensore, un robot per la cucina, un bancomat.*

Un automa a stati finiti è formato da:

- T = {insieme discreto dei tempi}
- I = {insieme finito delle variabili d'ingresso}
- VI = {insieme finito delle variazioni degli ingressi, di possibili valori dei vari ingressi}
- U = {insieme finito delle variabili d'uscita}
- VU = {insieme finito delle variazioni delle uscite, di possibili valori delle varie uscite}
- S = {insieme finito degli stati}

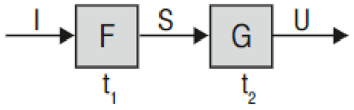
Gli automi a stati finiti

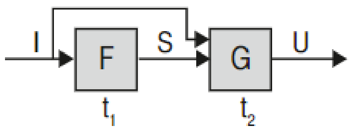
Automa proprio (di Moore)

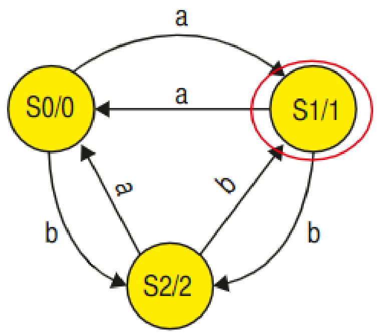
quando la funzione di trasformazione dipende solo dai valori assunti dallo stato ($U_{m(t)} = g_m\ (S_{(t)})$), cioè quando l'uscita dipende solo dallo stato nello stesso istante.

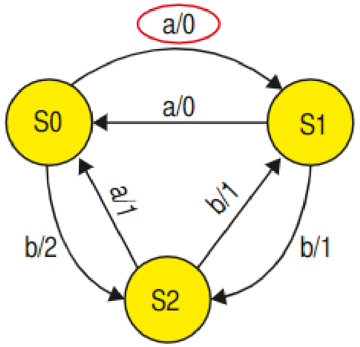
Automa improprio (di Mealy)

quando la funzione di trasformazione dipende sia dal valore dello stato sia da quelli assunti dall'ingresso allo stesso istante ($U_{m(t)} = g_m\ (S_{(t)}),\ I_{n(t)}$), cioè quando l'uscita dipende sia dall'ingresso sia dallo stato nello stesso istante

<table>
<tr><th></th><th>AUTOMA DI MOORE</th><th>AUTOMA DI MEALY</th></tr>
<tr><td>Significato</td><td>

Quando l'automa riceve un segnale di ingresso, effettua dapprima la transizione verso lo stato futuro e poi calcola l'uscita in funzione del nuovo stato acquisito.</td><td>

Quando l'automa riceve un segnale di ingresso viene calcolata prima l'uscita e poi lo stato futuro dell'automa stesso.</td></tr>
<tr><td>Rappresentazione tabellare mediante tabella di transizione</td><td>
<table>
<tr><th></th><th></th><th colspan="2">Variaz. ingressi</th></tr>
<tr><th></th><th></th><th>a</th><th>b</th></tr>
<tr><td rowspan="3">Stati/Uscite</td><td>S0/0</td><td>S1/1</td><td>S2/2</td></tr>
<tr><td>S1/1</td><td>S0/0</td><td>S2/2</td></tr>
<tr><td>S2/2</td><td>S0/0</td><td>S1/1</td></tr>
</table>
</td><td>
<table>
<tr><th></th><th></th><th colspan="2">Variaz. ingressi</th></tr>
<tr><th></th><th></th><th>a</th><th>b</th></tr>
<tr><td rowspan="3">Stati</td><td>S0</td><td>S1/ 0</td><td>S2 / 2</td></tr>
<tr><td>S1</td><td>S0 / 0</td><td>S2 / 1</td></tr>
<tr><td>S2</td><td>S0/ 1</td><td>S1 / 1</td></tr>
</table>
</td></tr>
<tr><td>Rappresentazione grafica mediante diagramma degli stati</td><td>Dal punto di vista grafico la rappresentazione di questo automa avviene nel modo seguente:


Stato e uscita sono rappresentati nello stesso riquadro, in quanto uno è la diretta conseguenza dell'altro.</td><td>Dal punto di vista grafico la rappresentazione di questo automa avviene nel modo seguente:


L'arco che porta da uno stato all'altro contiene sia l'ingresso applicato sia l'uscita ottenuta.</td></tr>
</table>

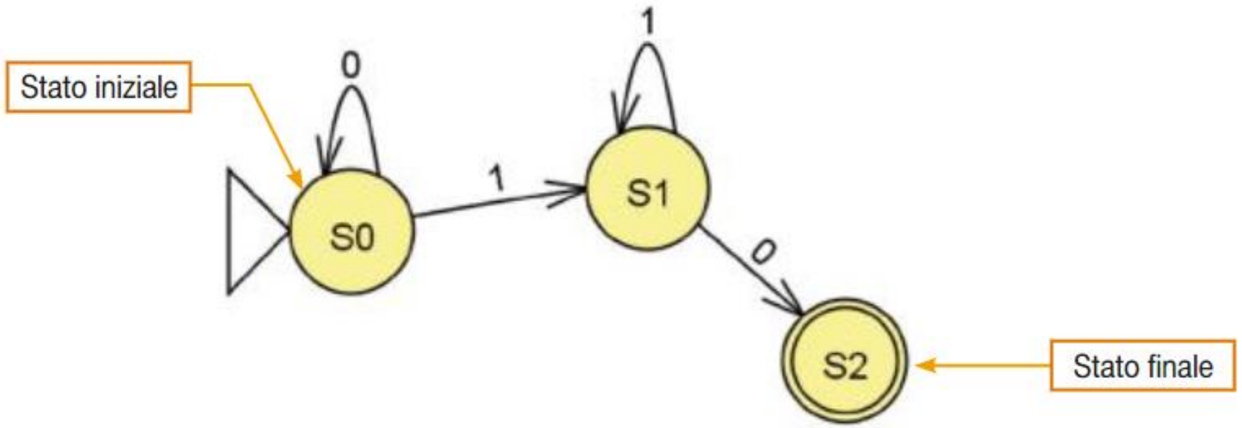
Proprietà degli automi

Stato iniziale

Quello in cui si trova il sistema all'inizio del suo funzionamento, quando non è ancora pervenuto alcun valore in ingresso. È rappresentato da un nodo con una singola freccia o cuneo entrante.

Stato finale

Quello in cui si trova il sistema quando termina con successo l'esecuzione. È spesso rappresentato con un doppio cerchio e non è individuabile in tutti gli automi. Può essere stato finale anche quando l'automa non è più in grado di passare a uno stato successivo, per qualsiasi valore applicato all'ingresso.



Stato di equilibrio

Quando l'insieme delle variazioni degli ingressi mantiene l'automa sempre nello stesso stato.

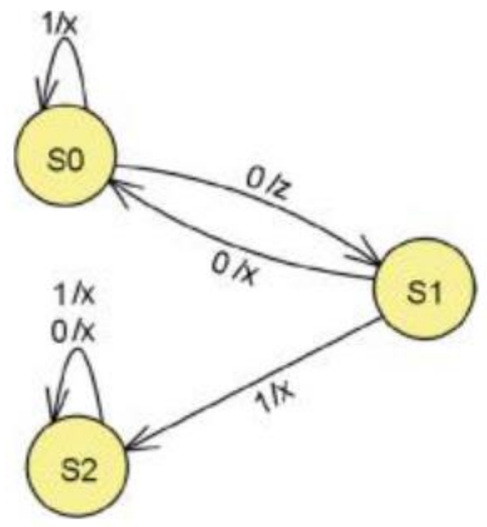
Uscita di equilibrio

Si ottiene se esistono uno stato iniziale e una sequenza di ingressi tali da mantenere l'uscita sempre allo stesso valore.

<u>*Esempio:*</u>

Nell'esempio accanto (automa di Mealy) possiamo notare che:

- *lo stato S0 è di equilibrio se riceve ingresso 1. S0 non ha uscite in equilibrio;*
- *lo stato S1 ha l'uscita X di equilibrio. S1 non è di equilibro per nessun valore di ingresso;*
- *lo stato S2 è di equilibrio se riceve in ingresso 0 o 1 e quindi è anche stato finale. S2 ha anche l'uscita X di equilibrio.*

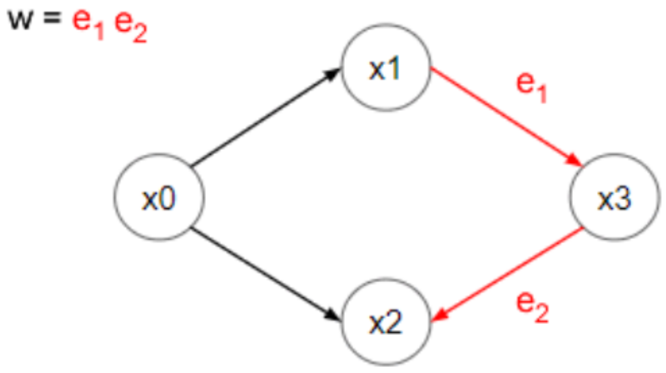


Stati indistinguibili

Due stati si dicono indistinguibili se, per ingressi uguali, possiedono le stesse uscite e lo stesso stato di arrivo

Stato raggiungibile

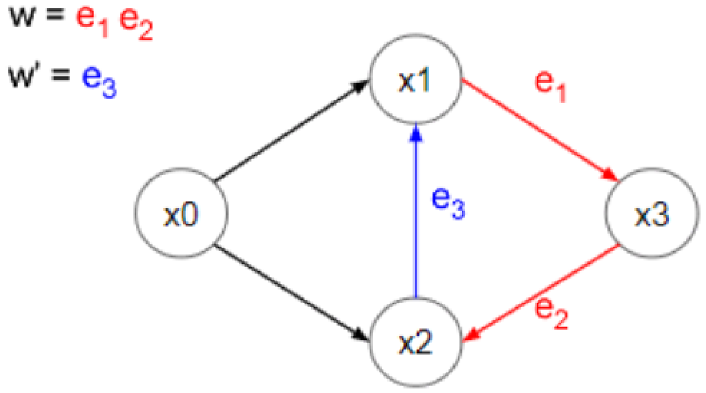
In un automa uno stato x_2 è raggiungibile dallo stato x_1 se è definita la transizione $f(x_1,w)=x_2$ con una parola w, ossia con una serie di eventi.
La raggiungibilità dello stato equivale all'esistenza di un cammino orientato dal nodo x_1 al nodo x_2 in un grafo orientato (**digrafo**).



Si parla di **raggiungibilità diretta** quando si passa direttamente dallo stato x_1 allo stato x_2 o di **raggiungibilità indiretta** quando si passa dallo stato x_1 allo stato x_2 attraverso stati intermedi.

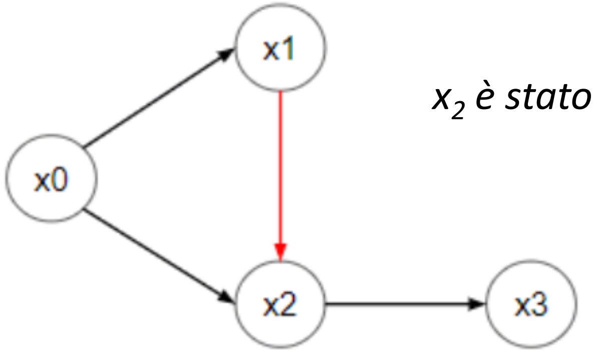
Stato co-raggiungibile

In un automa uno stato x_2 è detto co-raggiungibile verso x_1, se è definita una transizione $\delta(x_2,w')= x_1$ con una parola w‘ (produzione inversa).
In un digrafo questo vuol dire che esiste un cammino orientato da x_2 a x_1 (co-raggiungibilità).
Il cammino w' congiunge x_2 con x_1. Quindi x_2 è co-raggiungibile verso x_1.



Stato bloccante

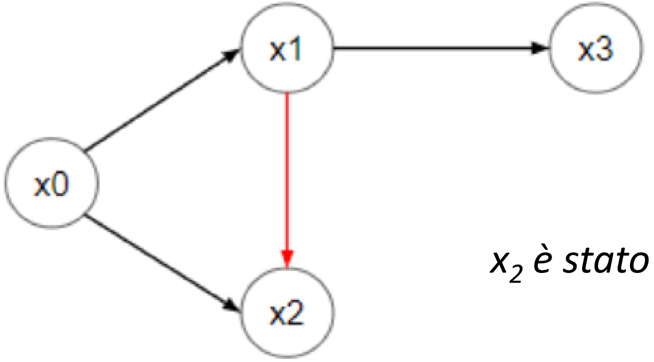
In un digrafo lo stato bloccante è individuato da un cammino orientato da x_1 a x_2 e dalla mancanza di un cammino inverso da x_2 a x_1.



x_2 è stato bloccante

Stato morto

Nel caso in cui lo stato bloccante non abbia altri archi in uscita è anche uno stato morto.



x_2 è stato morto

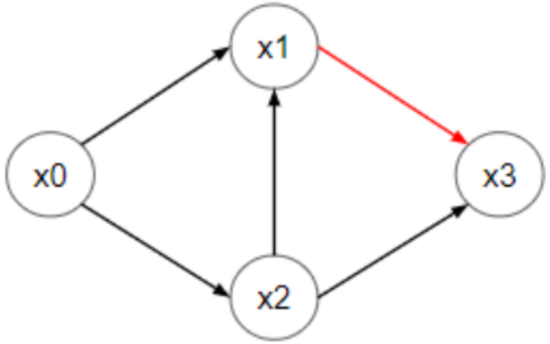
La non raggiungibilità dello stato

Lo stato x_2 è detto non raggiungibile da x_1 se non esiste una parola w (sequenza di eventi) in grado di portare l'automa dallo stato x_1 allo stato x_2.
In questo caso la transizione $x_2=\delta(x_1,w)$ non è definita:

$$x_2=\delta(x_1,w)!$$

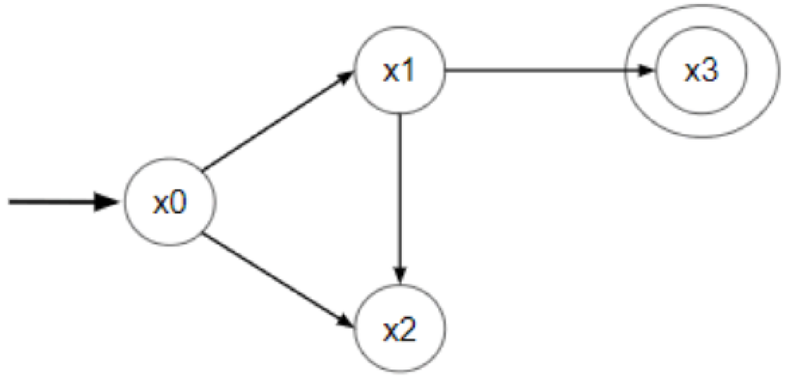
In un digrafo questo equivale all'assenza di un cammino orientato dal nodo x_1 al nodo x_2

Nel digrafo sotto, lo stato x_2 è non raggiungibile dallo stato x_1 perché la direzione degli archi è opposta. E così via.



Automa raggiungibile

È un automa in cui tutti gli stati sono raggiungibili. Rappresentando l'automa raggiungibile con un grafo orientato (digrafo), tutti i nodi sono raggiungibili (hanno frecce entranti).

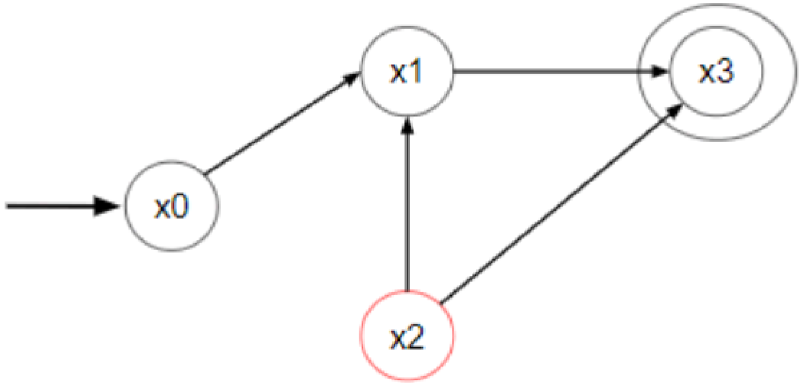


Ogni nodo del digrafo è uno stato dell'automa a stati finiti. Ogni arco è una transizione da uno stato a un altro.

A cosa serve la raggiungibilità?
permette di capire in quali stati può trovarsi l'automa a partire da uno stato (stato futuro in funzione dello stato di partenza)

Automa non raggiungibile

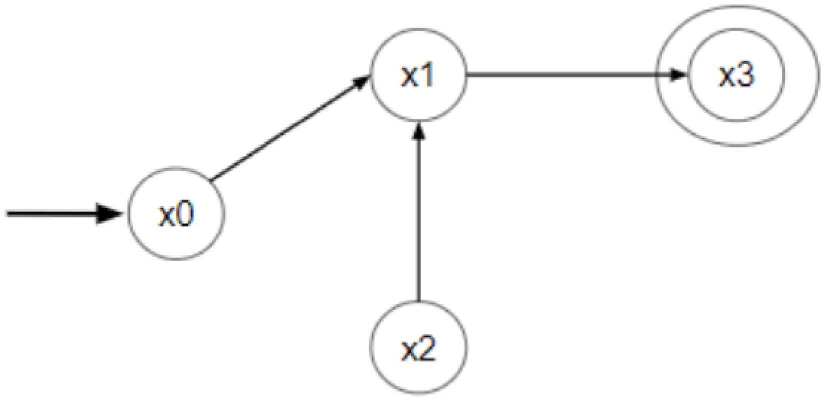
Se uno o più stati non sono raggiungibili, l'automa è detto automa non raggiungibile.



Ad esempio, nel digrafo riportato sopra, lo stato x_2 non è raggiungibile da x_0 e x_1 per cui l'automa nella sua interezza non è raggiungibile.

Automa co-raggiungibile

Un automa co-raggiungibile ha tutti gli stati che sono co-raggiungibili (in un digrafo tutti gli stati devono avere delle frecce uscenti, escluso eventualmente il nodo di uscita)



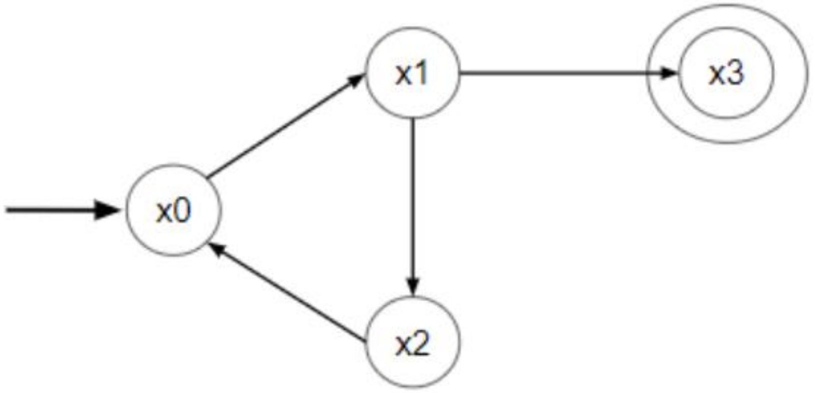
Nota

Un automa co-raggiungibile potrebbe essere raggiungibile o non raggiungibile.
Ad esempio, nel grafo di sopra gli stati x_1 e x_2 sono co-raggiungibili ma x_2 non è raggiungibile. Quindi, l'automa è co-raggiungibile ma non raggiungibile.

Automa rifinito

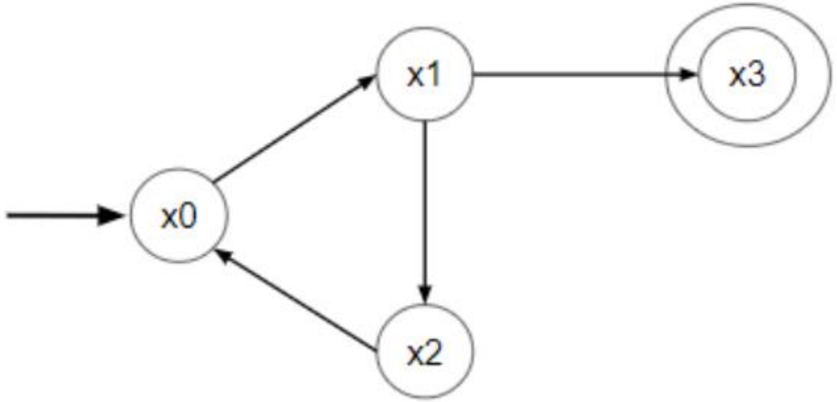
Un automa rifinito ha tutti i nodi sia raggiungibili che co-raggiungibili.

Nell'esempio sotto, escludendo il nodo iniziale (x_0) e finale (x_3), tutti i nodi sono sia raggiungibili che co-raggiungibili



Automa reversibile

Un automa è reversibile se ogni nodo è raggiungibile dallo stato iniziale e co-raggiungibile verso lo stato iniziale.



Nell'esempio sopra, i nodi x_1 e x_2 sono sia:

- *raggiungibili da x_0 stato iniziale ($\exists$ un percorso orientato da x_0 a x_1 e da x_0 a x_2) e*
- *co-raggiungibili verso x_0 ($\exists$ un percorso da x_1 a x_0 e da x_2 a x_0)*

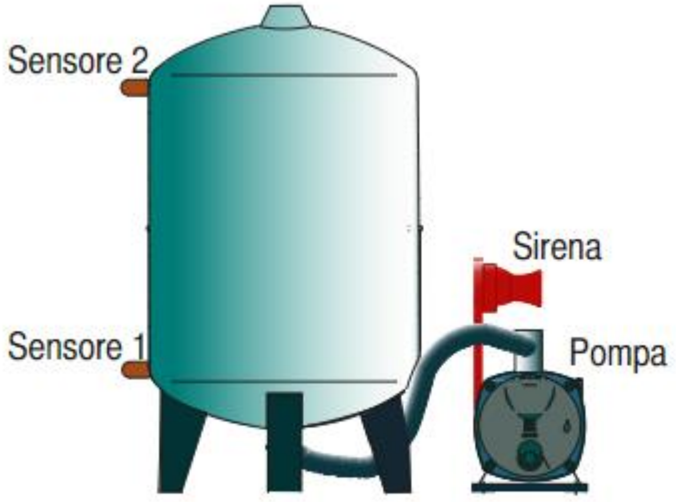
A cosa serve la reversibilità?

La reversibilità equivale a dire che il sistema può sempre essere reinizializzato (dal momento che l'inizializzazione avviene nello stato iniziale).

es.1 **AUTOMA SERBATOIO**

In questo esempio vogliamo realizzare un sistema automatico (automa) per il controllo del riempimento di un serbatoio di carburante.

La verifica del livello raggiunto dal liquido versato avviene per mezzo di due **sensori**, uno posto in alto e uno sul fondo della vasca contenitrice, che possiamo definire **Ingressi** del sistema.
I sensori inviano un segnale di tipo 1 o 0 a seconda della presenza di carburante o meno in corrispondenza degli stessi.



L'**uscita** del sistema è invece rappresentata dalla **pompa**, in grado di versare il carburante, e dalla sirena di allarme. La pompa può assumere tre valori:

- 2 per pompare carburante all'interno del serbatoio,
- 1 per far uscire il carburante in eccesso e
- 0 quando è ferma.

L'**uscita Sirena** indica:

- una condizione di errore nella quale il sensore posto in alto è acceso (1) mentre il sensore posto in basso è spento (0). Quando viene raggiunta questa situazione l'uscita Sirena va a 1 e mantiene questo stato
- in caso contrario resta spenta (0)

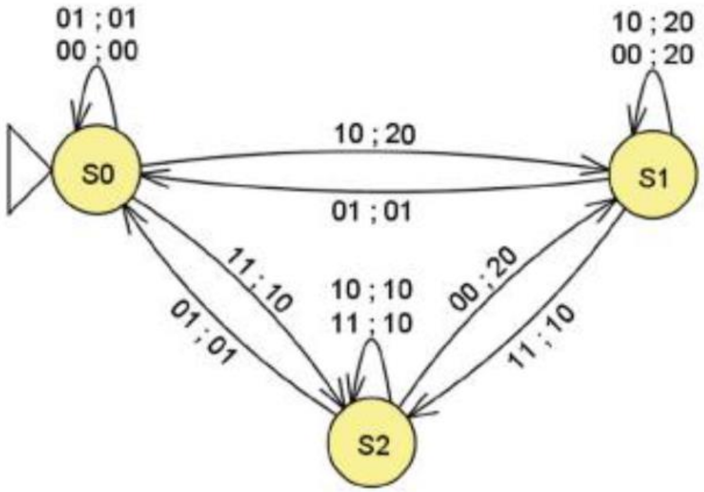
Valori degli ingressi, uscite e stati dell'automa

- I = Sensore 1 e Sensore 2
- VI = {00, 01, 10, 11} = Vuoto, Carburante in alto ma non in basso (impossibile), Carburante in basso ma non in alto, Pieno
- U = Pompa, Sirena
- VU = {00, 01, 10, 20, 11, 21} = pompa e sirena ferme, pompa ferma e sirena attiva, pompa fuori carburante, pompa dentro carburante, situazione impossibile, situazione impossibile
- S= {S0, S1, S2} = {Fermo, Riempimento, Svuotamento}

Tabella di transizione degli stati

STATI	INGRESSI			
	00	01	10	11
S0	S0/00	S0/01	S1/20	S2/10
S1	S1/20	S0/01	S1/20	S2/10
S2	S1/20	S0/01	S2/10	S2/10

Diagramma degli stati (Mealy)

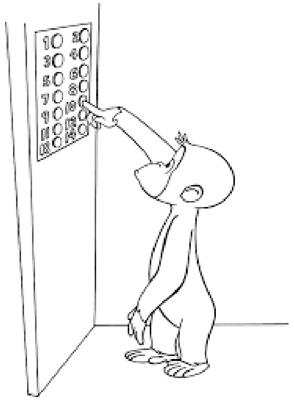


es.2 **AUTOMA ASCENSORE A TRE PIANI**

In questo esempio vogliamo realizzare un sistema automatico (automa) per il controllo di un ascensore a tre piani. La pulsantiera posta nella cabina dell'ascensore accetta tre possibili valori (0, 1, 2), l'uscita è rappresentata dal motore che può far salire, scendere o stare fermo l'ascensore stesso (2, 1, 0), mentre lo stato è indicato dal piano in cui si trova (0, 1, 2).

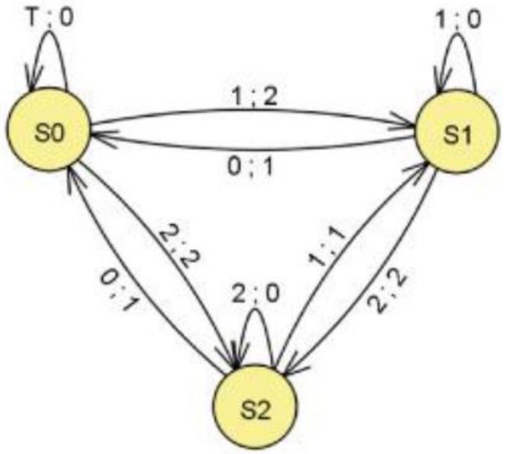
<u>Valori degli ingressi, uscite e stati dell'automa:</u>

- I = tasto richiesta
- VI = {T, 1, 2} = Terra, Primo, Secondo
- U = azione motore
- VU = {2, 1, 0} = Sale, Scende, Fermo
- S = {S0, S1, S2} = Piano terra, Primo piano, Secondo piano



In questo caso abbiamo omesso volutamente la pulsantiera di richiesta per semplificare l'esercizio e si ipotizza ci sia un unico pulsante/tasto di richiesta per selezionare i piani

STATI	INGRESSI		
	T	1	2
S0	S0/0	S1/2	S1/2
S1	S0/1	S1/0	S2/2
S2	S0/1	S1/1	S2/0



es.3 **AUTOMA DI MOORE: SEMAFORO**

In questo esempio vogliamo realizzare un automa in grado di gestire un semaforo stradale. Un semaforo può trovarsi in uno dei seguenti stati:

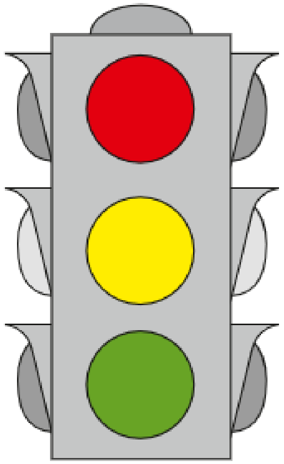
S = {G, R, V, L} = {Giallo, Rosso, Verde, Lampeggiante}

In ingresso al sistema quindi possiamo avere due segnali possibili, dati da un ipotetico sensore crepuscolare in grado di leggere la luminosità esterna:

I = {Giorno, Notte}

Le uscite del sistema sono le lampadine del semaforo che sono strettamente associate allo stato del semaforo: ad ogni stato corrisponde una sola uscita e ad ogni uscita corrisponde un solo stato.
Questo esempio descrive quindi un sistema in cui l'uscita dipende solo dallo stato corrente; un automa di questo tipo è detto automa di Moore.
Le uscite dell'automa sono sempre conseguenza dello stato corrente, vediamole nel dettaglio:

U = {L, R, G, V} = {Lampeggiante, Rosso, Giallo, Verde}

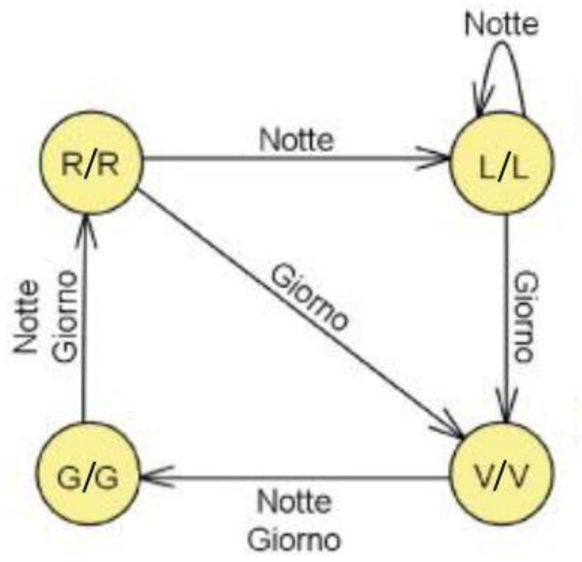


Durante il giorno il semaforo segue una sequenza di stati che è stata impostata, mentre durante la notte il semaforo entra in stato di lampeggiamento. Quando il sensore crepuscolare rileva il buio, l'automa deve attendere il primo rosso per passare allo stato lampeggiante, rimanendo nello stato lampeggiante fino alla prossima transizione notte/giorno.

Vediamo a lato tutte le possibili transizioni degli stati del semaforo in funzione degli ingressi e dello stato precedente, riassunti nella tabella seguente.

	GIORNO				NOTTE			
Stato precedente (S0)	Verde	Giallo	Rosso	Lampeggiante	Verde	Giallo	Rosso	Lampeggiante
Stato successivo (S1)	Giallo	Rosso	Verde	Verde	Giallo	Rosso	Lampeggiante	Lampeggiante

Diagramma degli stati:



es.4 **AUTOMA PARCHEGGIO**

Vogliamo progettare un sistema di gestione di un parcheggio pubblico mediante un automa. Le auto in ingresso e in uscita dal parcheggio possono essere conteggiate mediante un sensore ottico (fotocellula).
La sbarra in entrata si abbassa per limitare l'accesso alle auto qualora la capienza sia stata raggiunta dalle auto presenti.

Modello matematico:

- I = {E1, E2} = fotocellula all'entrata, fotocellula all'uscita (segnali che uso per controllare l'automa parcheggio: cause dello stato)
- VI = {00, 01, 10, 11} = nessun veicolo, esce veicolo, entra veicolo, entra ed esce veicolo

- U = {S} = sbarra di accesso (ciò che voglio controllare con il mio automa: effetto dello stato)
- VU = {0, 1} = sbarra su, sbarra giù

Lo **stato** del sistema può essere rappresentato dal **numero di veicoli presenti nel parcheggio**:

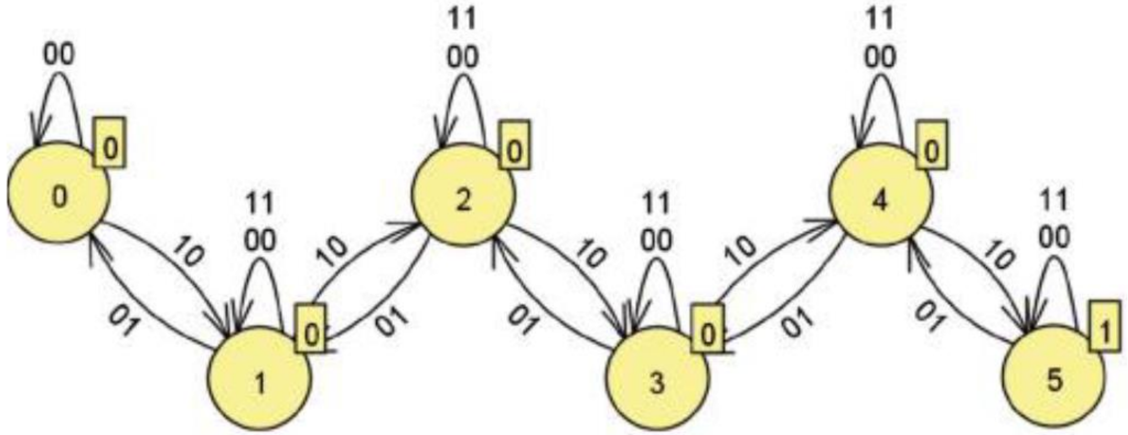
- S = {N} = supponendo di avere una capienza massima di 5 veicoli nel parcheggio

Tabella degli stati:

STATI	INGRESSI				USCITA
	00	01	10	11	
0	0	err.	1	err.	0
1	1	0	2	1	0
2	2	1	3	2	0
3	3	2	4	3	0
4	4	3	5	4	0
5	5	4	err.	5	1

Diagramma degli stati:

L'automa considerato è di tipo **Moore** in quanto l'uscita dipende solo dallo stato nello stesso istante (è diretta conseguenza dello stato)



es.5 **AUTOMA FRULLATORE**

Un frullatore elettrico per poter funzionare correttamente necessita di essere acceso e della presenza del tappo sul bicchiere. Progettare un automa a stati finiti (tabella transizioni + grafo di Mealy o Moore) per controllare il funzionamento del frullatore. Il segnale d'uscita resta basso, a meno che il frullatore non sia acceso e non sia presente il tappo.
L'insieme degli ingressi è costituito da: MT: Metti Tappo, TT: Togli Tappo, AF: Accendi Frullatore, SF: Spegni Frullatore
L'insieme delle uscite è costituito da: ON: Frullatore funzionante, OFF: Frullatore non funzionante
Lo stato iniziale è così caratterizzato: S0 = Senza Tappo e Frullatore Spento

<u>Modello matematico:</u>

- I = {tappo, accensione}
- VI = {MT, TT} , {AF, SF} = MT: Metti Tappo, TT: Togli Tappo, AF: Accendi Frullatore, SF: Spegni Frullatore
  = {00,01,10,11} = possibili combinazioni degli ingressi, dove 00=TT+SF, 01=TT+AF, 10=MT+SF, 11=MT+AF

- U = {funzionamento frullatore}
- VU = {ON, OFF} = ON: Frullatore funzionante, OFF: Frullatore non funzionante

Gli **stati** del sistema possono essere:

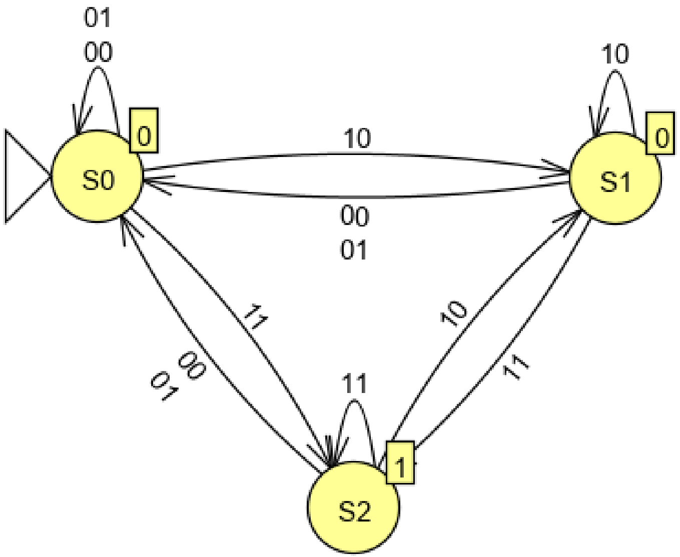
- S = {S0, S1, S2} S0 = Senza Tappo e Frullatore Spento
  S1 = Con Tappo e Frullatore Spento
  S2 = Con Tappo e Frullatore Acceso
  (non può esserci uno stato con frullatore acceso senza tappo)

Tabella degli stati:

Stati	Ingressi			
	00	01	10	11
S0	S0/OFF	S0/OFF	S1/OFF	S2/ON
S1	S0/OFF	S0/OFF	S1/OFF	S2/ON
S2	S0/OFF	S0/OFF	S1/OFF	S2/ON

Diagramma degli stati:

L'automa considerato è di tipo **Moore** in quanto l'uscita (essere acceso o spento il frullatore) dipende solo dallo stato nello stesso istante (avere tappo ed aver premuto pulsante di accensione)



es.6 **AUTOMA DI MOORE e MEALY: DISTRIBUTORE DI BIBITE**

Supponiamo di avere un distributore di bibite che accetta monete da 0.50€ e 1.00€ e la bibita viene erogata per 1.50€

Vediamo il caso di distributore senza resto e distributore con resto

Distributore senza resto

- I = moneta
- VI = { 0.50 , 1.00 }
- U = lattina
- VU = { - , ok }
- S = {0.00 , 0.50, 1.00}

	0.50	**1.00**
0.00	0.50/-	1.00/-
0.50	1.00/-	0.00/ok
1.00	0.00/ok	0.50/ok

Distributore con resto

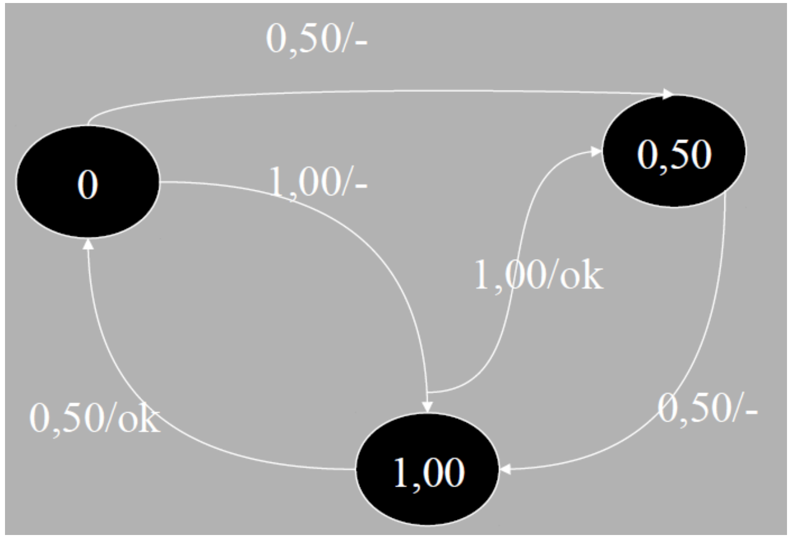
Quando non ho ancora raggiunto la quota per acquistare una lattina, non posso neppure calcolare il resto (non è come avere resto 0)

- I = moneta
- VI = { 0.50 , 1.00 }
- U = {lattina, resto}
- VU = { - , ok } , { - , 0 , 50}
- S = {0.00 , 0.50, 1.00}

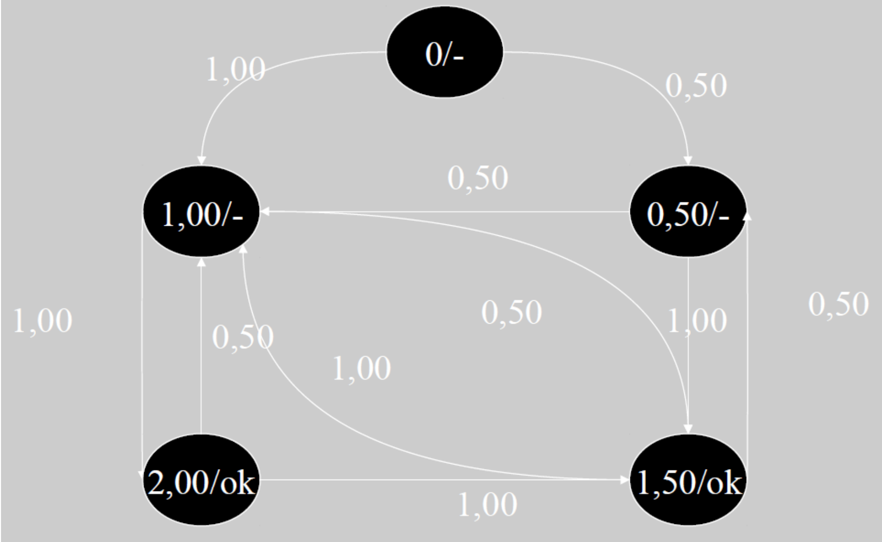
	0.50	**1.00**
0.00	0.50/-,-	1.00/-,-
0.50	1.00/-,-	0.00/ok,0
1.00	0.00/ok,0	0.00/ok,50

Prendendo in considerazione il solo distributore senza resto, vediamo le due diverse rappresentazioni

Grafo di Mealy



Grafo di Moore



Automa riconoscitore

Gli automi riconoscitori possono effettuare il **riconoscimento di una sequenza di valori** in ingresso, chiamato **pattern**.

Dopo aver ricevuto l'ultimo simbolo della sequenza di input restituiscono una risposta di tipo VERO/FALSO in base all'esito della verifica della sequenza stessa ricevuta. Lo **stato** indica l'avvenuto riconoscimento di **una parte della sequenza** cercata.

<u>Esempio:</u>
Gli automi riconoscitori trovano applicazione ad esempio nei telecomandi per la selezione dei canali della TV, oppure nei comandi per l'apertura a distanza di cancelli automatici.
Tramite il telecomando viene inviata all'apparecchiatura di controllo una sequenza seriale di bit, come livelli alti e bassi di onde elettromagnetiche, che rappresenta un valore numerico in forma binaria al quale corrisponde un determinato comando (per esempio apertura/chiusura). Il sistema di riconoscimento della sequenza è a tutti gli effetti un automa riconoscitore, in quanto invierà il comando al motore del cancello solo se la sequenza viene convalidata.

es. **AUTOMA RICONOSCITORE SEQUENZA**

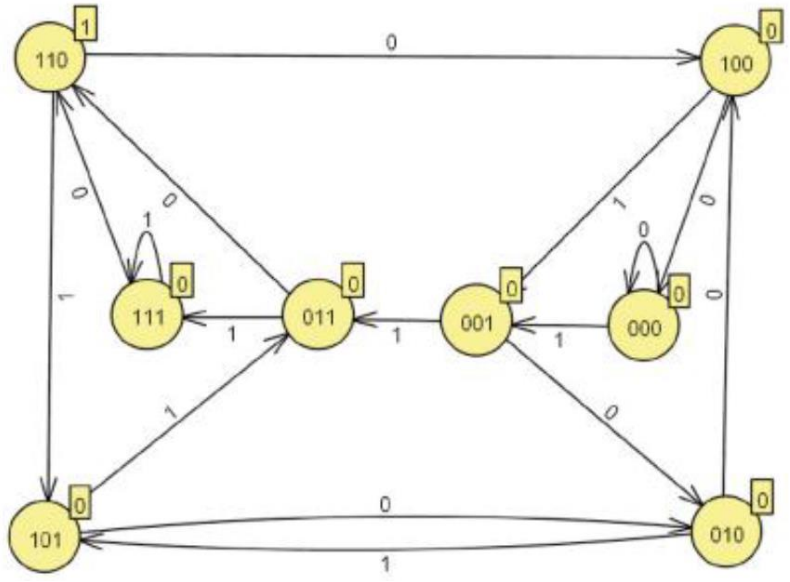
Vogliamo progettare un automa per il riconoscimento di una sequenza a tre bit, in grado di riconoscere la sequenza **110**.
Prima di tutto occorre stabilire gli **ingressi** possibili, in questo caso rappresentati dagli stati logici 0 e 1.
Quando la sequenza viene rilevata, il sistema restituisce in **uscita** il valore 1, cioè sequenza valida, in caso contrario 0, cioè sequenza non riconosciuta.
Immaginiamo che ogni nuovo bit in ingresso si posizioni a destra dei valori precedentemente pervenuti, e la configurazione a tre bit debba essere considerata ogni volta che viene letto in ingresso un nuovo bit. In tal modo appena la configurazione di tre bit consecutivi è quella desiderata viene inviato in uscita il valore di successo (1).

Tabella degli stati:

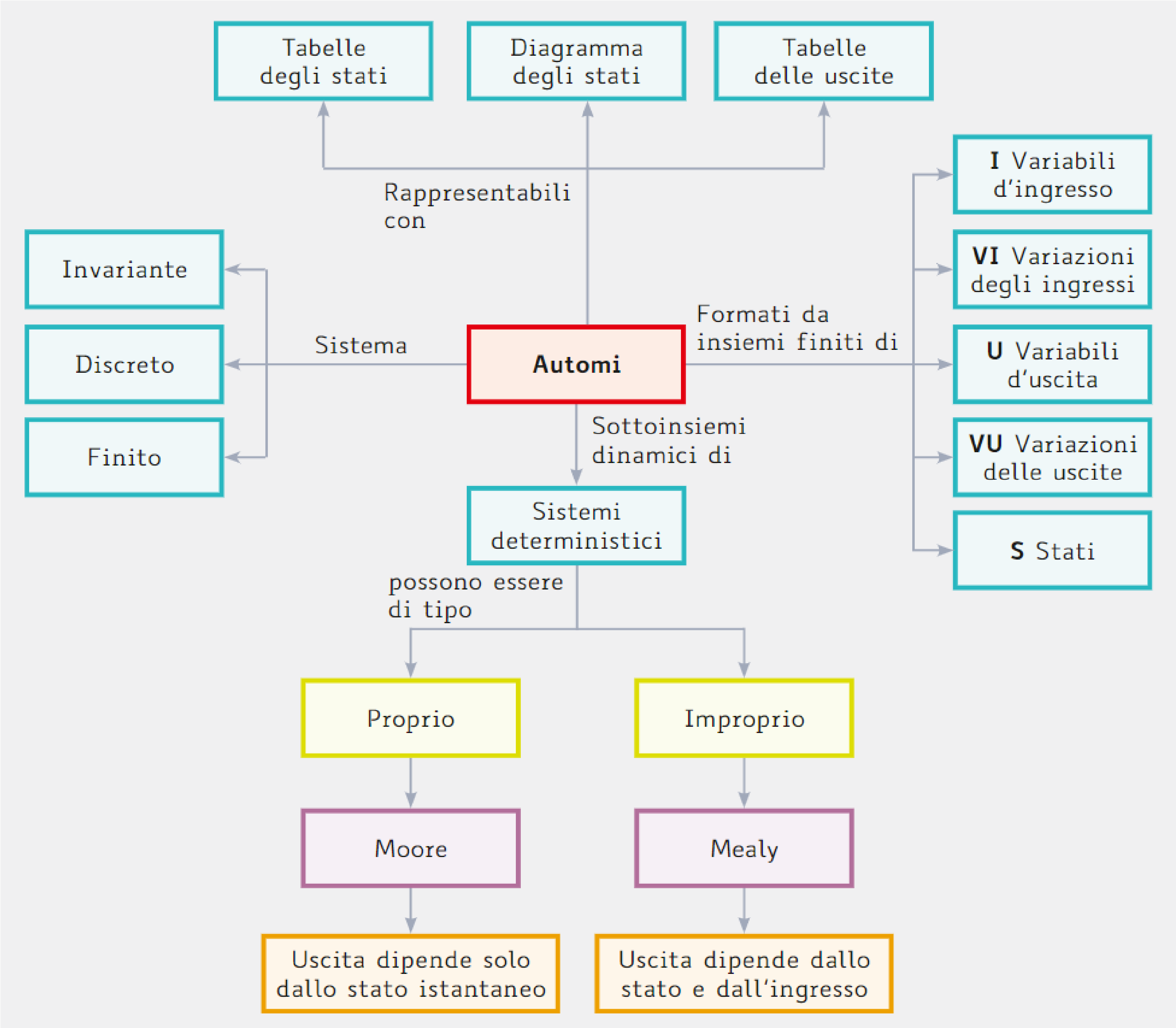
STATI	INGRESSI		USCITA
	0	1	
000	000	001	0
001	010	011	0
010	100	101	0
011	110	111	0
100	000	001	0
101	010	011	0
110	100	101	1
111	110	111	0

Diagramma degli stati:

L'automa considerato è di tipo **Moore** in quanto l'uscita dipende solo dallo stato nello stesso istante



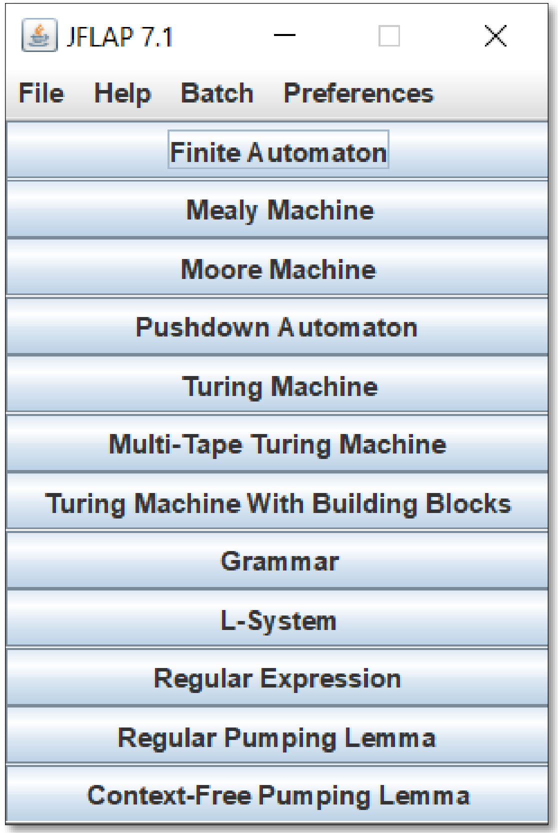
Mappa concettuale



Simulazione degli automi con JFLAP

Il progetto JFLAP (Java Formal Languages and Automata Theory) consente la simulazione di macchine a stati finiti e di Turing grazie ad un ambiente grafico.
Si tratta di un software progettato in Java e con licenza Creative Commons, quindi scaricabile ed utilizzabile gratuitamente ma non per scopo di lucro (www.jflap.org).

Eseguendo il file JFLAP.jar viene mandata in esecuzione la schermata principale del programma, dove possiamo scegliere l'operazione da svolgere (es. costruire un sistema con modello di Mealy o di Moore, etc.):

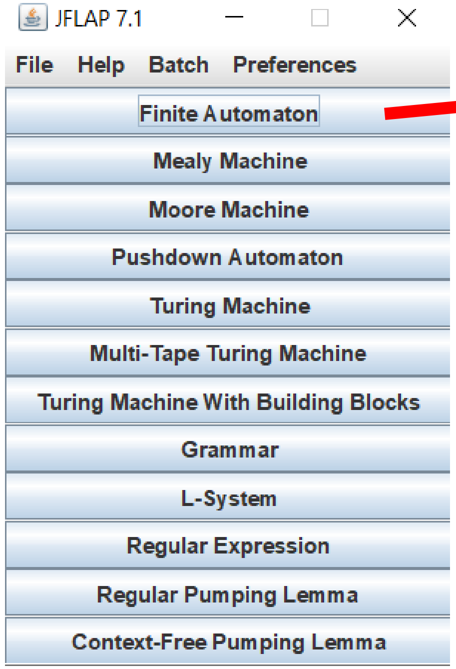


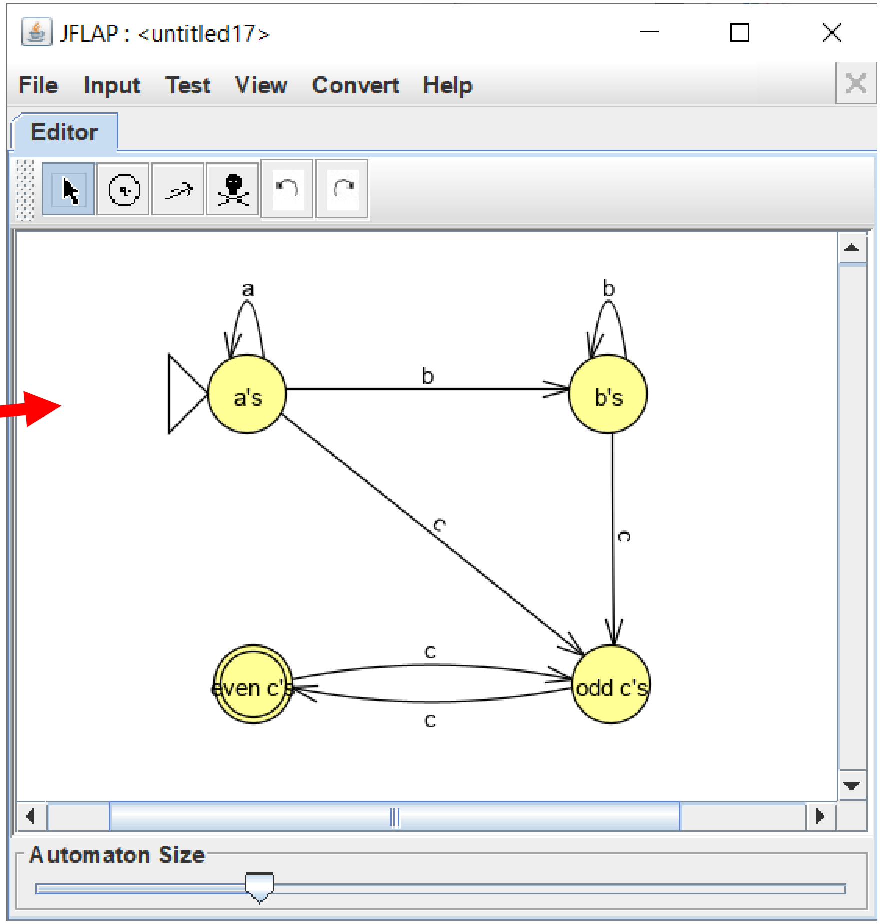
es. **AUTOMA RICONOSCITORE**

Creare un automa che riconosce una sequenza di questo tipo:

- un numero qualsiasi di «a»
- seguite da un numero qualsiasi di «b»
- seguite da un numero pari di «c»

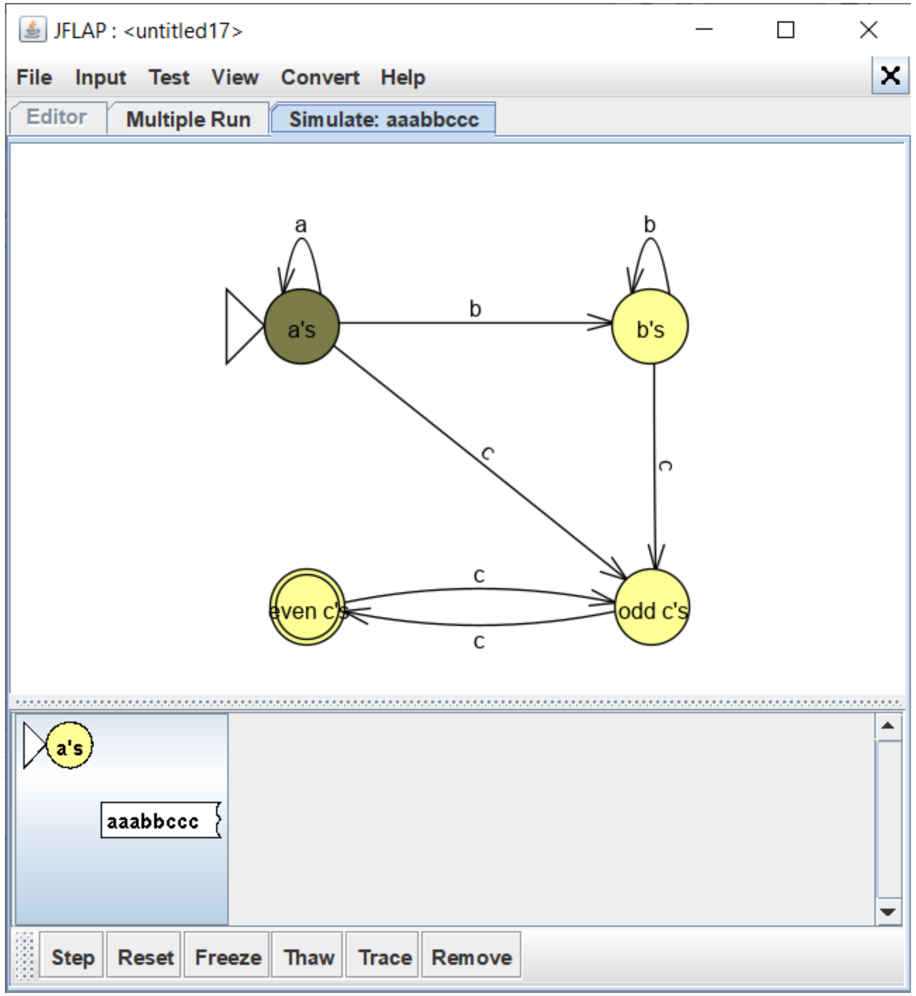
a…ab…b$\underbrace{\text{c……c}}_{\text{pari}}$



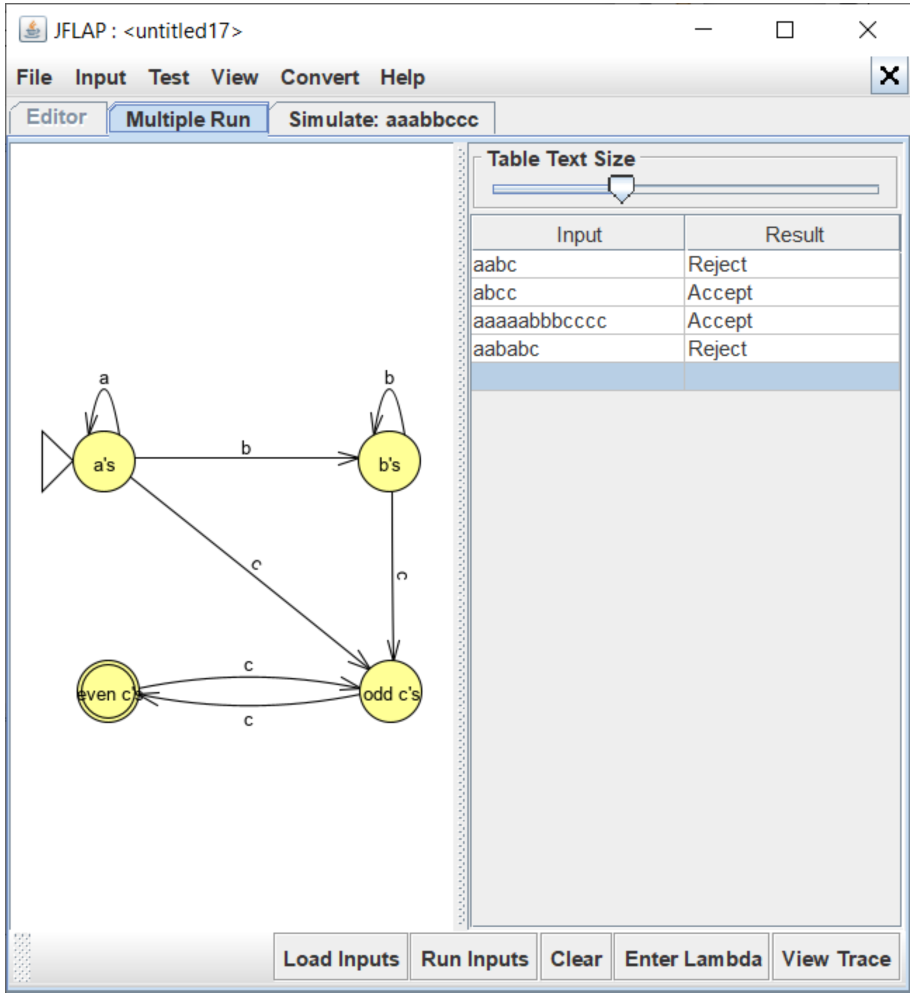


https://www.youtube.com/watch?v=cEvWgEHd0pE

Simulazione passo-passo (Input ⟶ Step with Closure)



Simulazione multipla (Input ⟶ Multiple run)



es. **SOMMATORE BINARIO SEQUENZIALE**

Vogliamo realizzare un automa che riceve in ingresso due sequenze di bit, dal meno significato al più significativo, quindi effettua l'addizione binaria usando l'algoritmo aritmetico della somma bit a bit con riporto.
Ad ogni passaggio l'automa deve memorizzare il risultato parziale attuale della somma dei due bit in input, oltre all'eventuale riporto che dovrà essere sommato ai bit successivi. Le combinazioni di risultato tra i due bit, oltre al riporto sono quindi quattro:

che daranno luogo a 4 stati:

Somma = 0 Riporto = 0 ⟶ S0
Somma = 0 Riporto = 1 ⟶ S2
Somma = 1 Riporto = 0 ⟶ S1
Somma = 1 Riporto = 1 ⟶ S3

Modello matematico:

- I = i due bit di input da sommare = $\{I_1, I_2\}$
- VI = possibili combinazioni di valori dei bit di input = {00, 01, 10, 11}

- U = somma = $\{U_1\}$
- VU = possibili valori della somma = {0, 1}

Lo **stato** del sistema è ciò che il sistema ricorda, ciò di cui deve tenere memoria, e in questo caso (come detto sopra) è il risultato parziale oltre al riporto. Per questo individuo 4 stati:

- S = {S0, S1, S2, S3}

Essendo le uscite diretta conseguenza dello stato, il modello che utilizziamo per questo sistema è quello di Moore.

Tabella degli stati:

STATI	INGRESSI			
	00	01	10	11
S0	S0/0	S1/1	S1/1	S2/0
S1	S0/0	S1/1	S1/1	S2/0
S2	S1/1	S2/0	S2/0	S3/1
S3	S1/1	S2/0	S2/0	S3/1

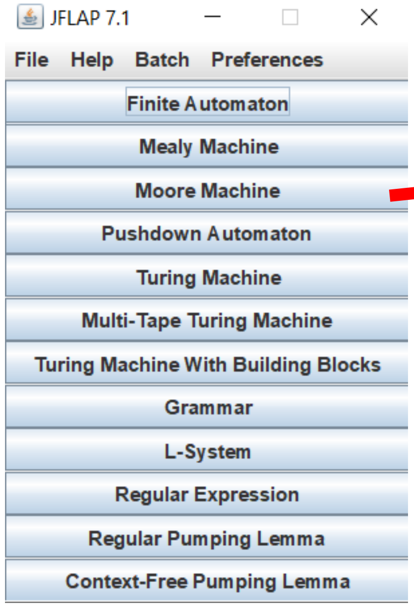
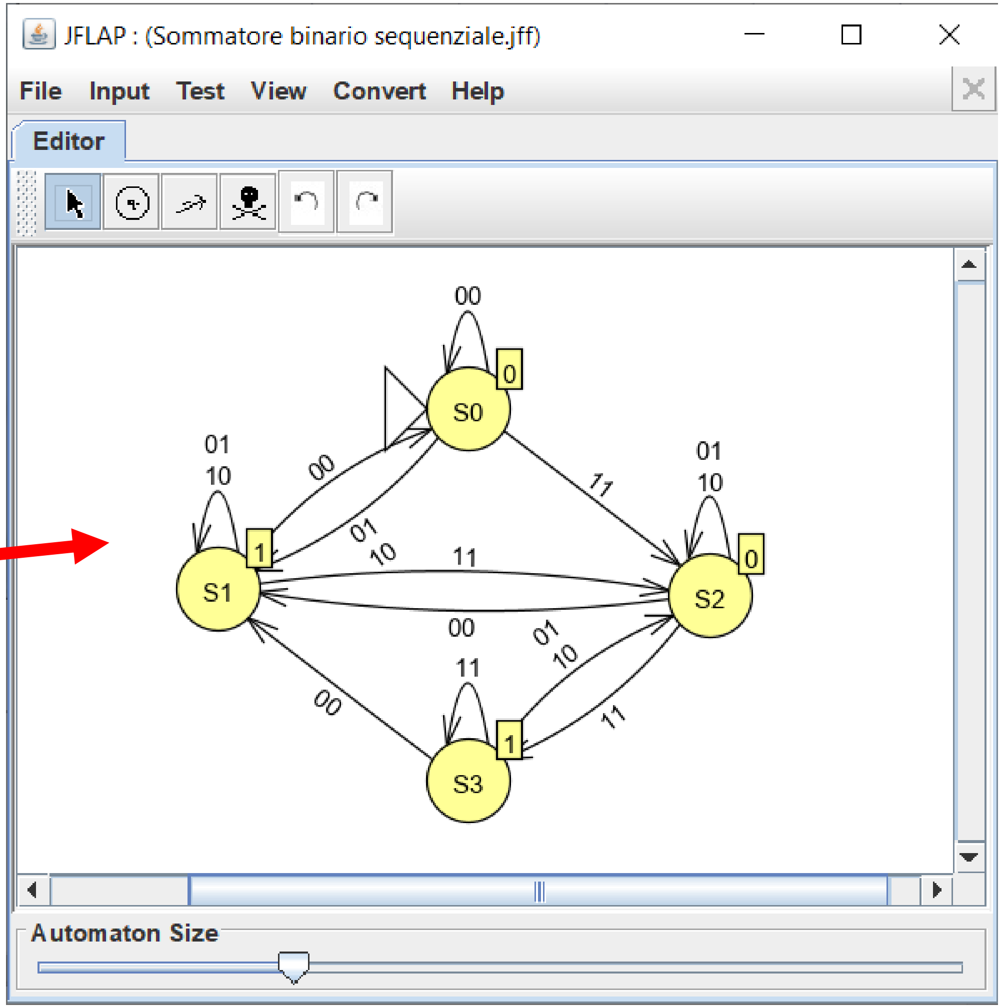
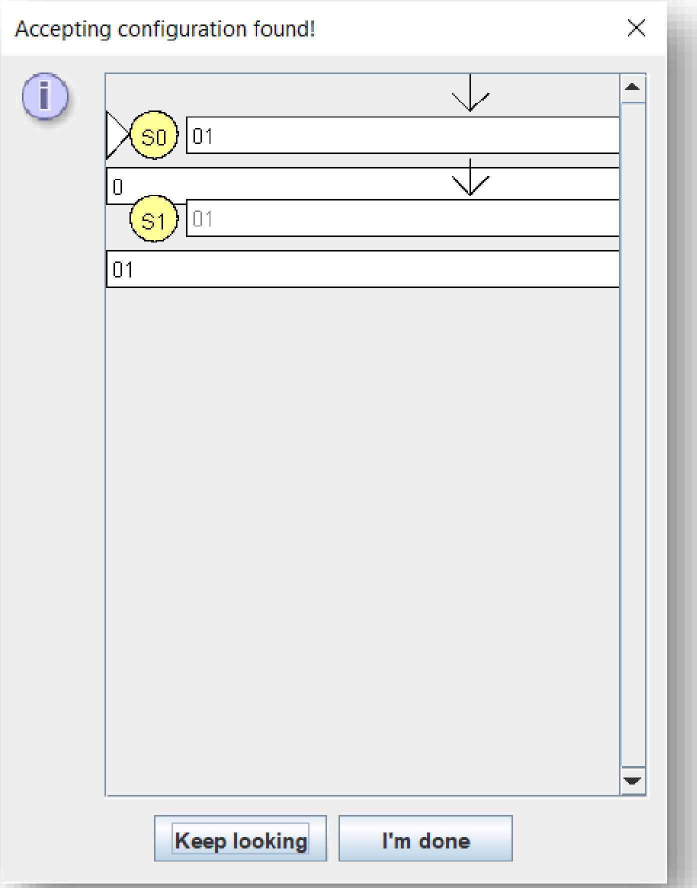
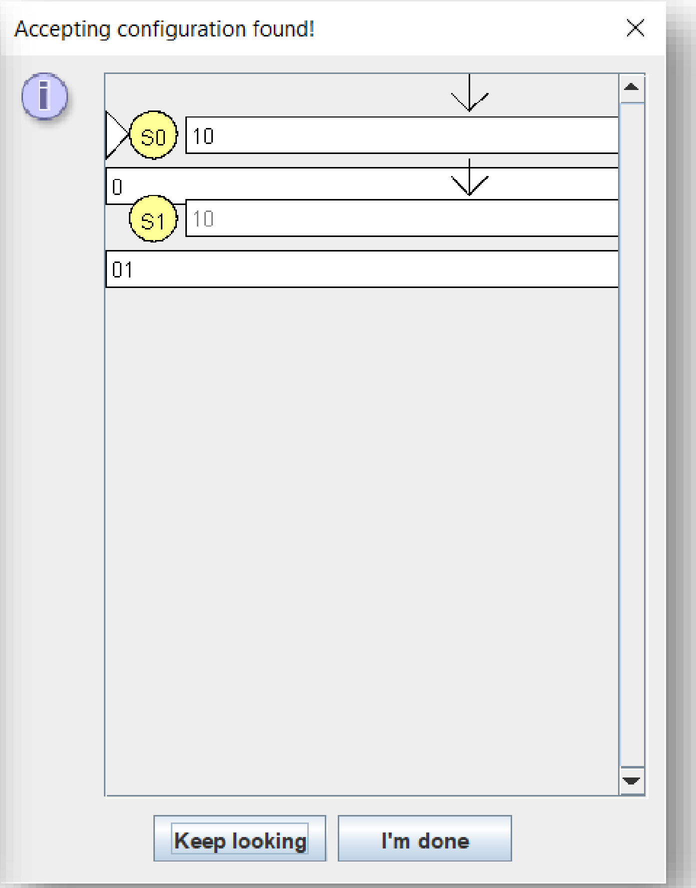


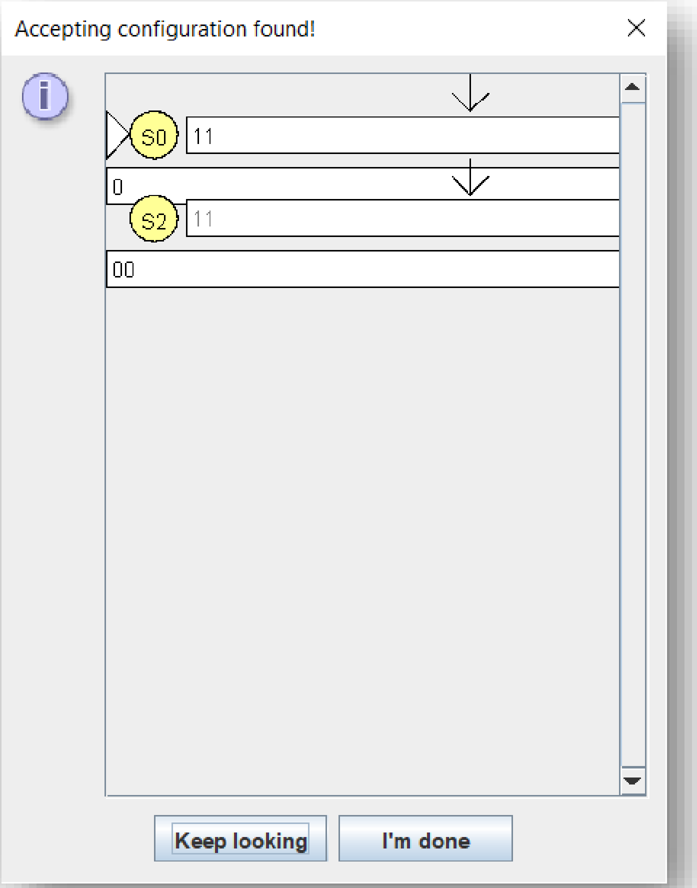
Diagramma degli stati:



Simulazione veloce (Input → Fast run)

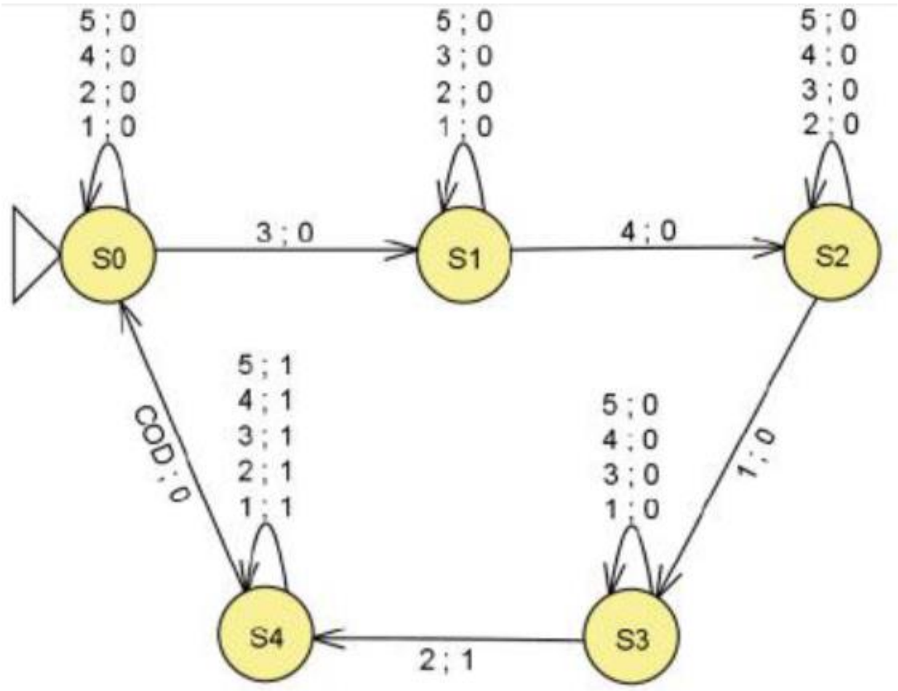






es. **TASTIERA A 5 TASTI**

Dato il seguente automa di Mealy, trasformalo nel corrispondente automa di Moore. L'automa seguente simula il funzionamento di una tastiera a 5 tasti (1, 2, 3, 4 e 5) dove viene digitato il codice a quattro cifre (che devi scoprire) che apre una cassaforte. Una volta che la cassaforte è aperta può essere premuto il tasto COD per richiuderla.



Utilizzando l'ambiente JFLAP simulane il funzionamento per scoprire la sequenza di apertura della cassaforte.