

RETI LOGICHE T

Ingegneria Informatica

Informazioni, programma e materiale didattico
www.vision.deis.unibo.it/corsi/retilogiche

Dispense e materiale didattico

R. Laschi, M. Prandini: "Reti Logiche", Esculapio, 2007
Morris Mano, Kime, «Reti Logiche», Prentice Hall

Ricevimento

Mercoledì ore 15-17 o in qualunque altro giorno previo
accordo via e-mail (federico.tombari@unibo.it)

Tutor: Marco Lippi (marco.lippi3@unibo.it)

Regolamento prove d'esame

In ogni A.A. sono previsti **6 appelli** d'esame, distribuiti in 3 sessioni:

sessione estiva (Giugno, Luglio)

sessione autunnale (Settembre)

sessione invernale (Gennaio, Febbraio)

L'esame consiste in una prova scritta che include uno o più esercizi

A richiesta dello studente, è possibile sostenere una prova orale facoltativa a perfezionamento della prova scritta

Programma

1. Introduzione
2. Codifica dell'informazione
3. Reti combinatorie
4. Reti sequenziali
5. Reti asincrone
6. Reti sincrone

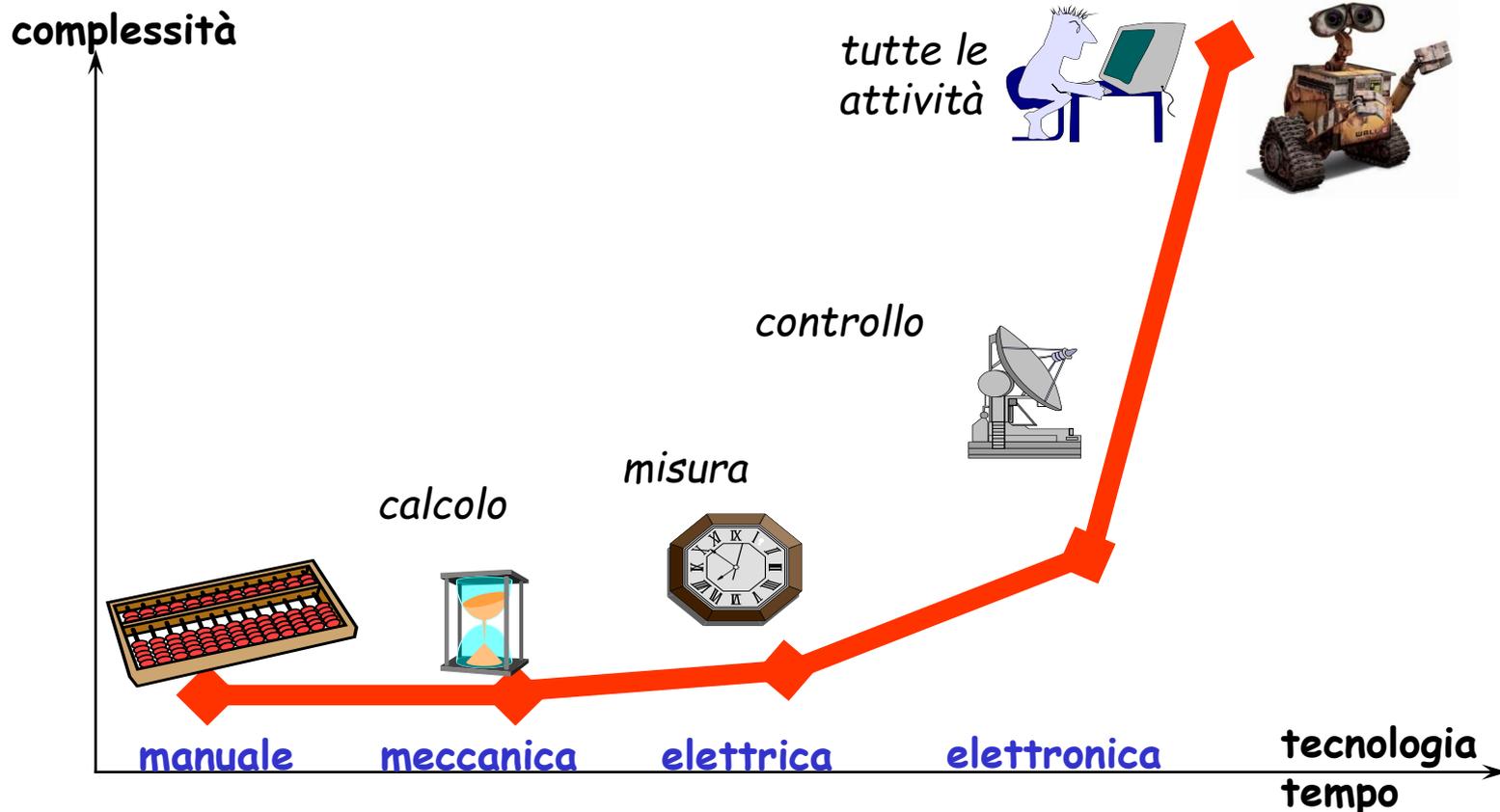
Capitolo 1

Introduzione

Reti Logiche T

3000 anni di storia e 4 tecnologie

- Le **macchine artificiali** hanno subito una enorme evoluzione
- Da strumenti per il calcolo a strumenti per la misura, a controllori di altri sistemi artificiali, a sistemi *general-purpose* e *intelligenti*
- Utilizzo di tecnologie sempre più evolute per ridurre *costo*, *consumi*, *ingombro* e aumentare la *velocità*

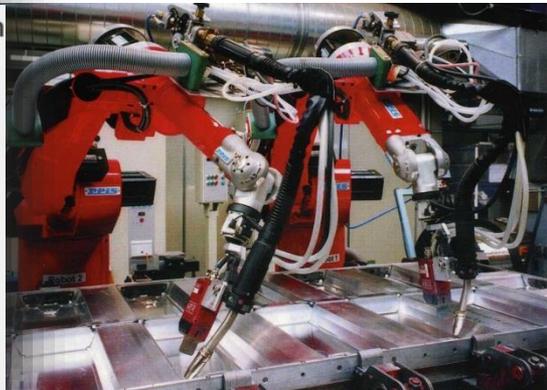


Macchine digitali

- **Macchine digitali:** Sistemi artificiali che:
 - impiegano grandezze fisiche *contraddistinte da un insieme discreto di valori "significativi" (segnali digitali, il mezzo)*
 - *per rappresentare, elaborare, comunicare informazioni (il fine)*
- **Digitale** (digit, cifra): relativo a sistemi basati su elementi di informazione **discreti**

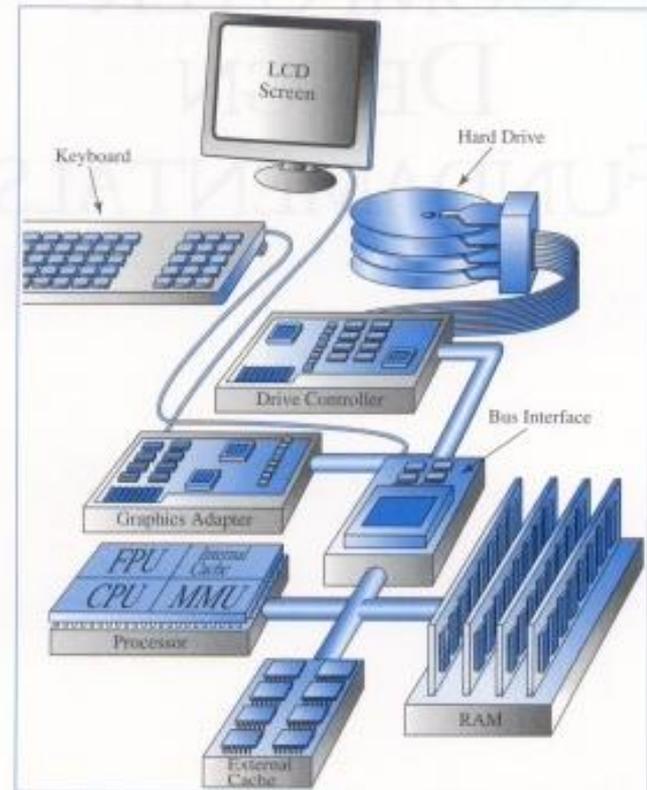
"Pervasive Digital Era"

- **Digital revolution:** passaggio dall'economia tradizionale ad una basata su tecnologie digitali (computer, www, sistemi avanzati per l'automazione, ..)
- **Macchine e tecnologie digitali sono ora pervasive**



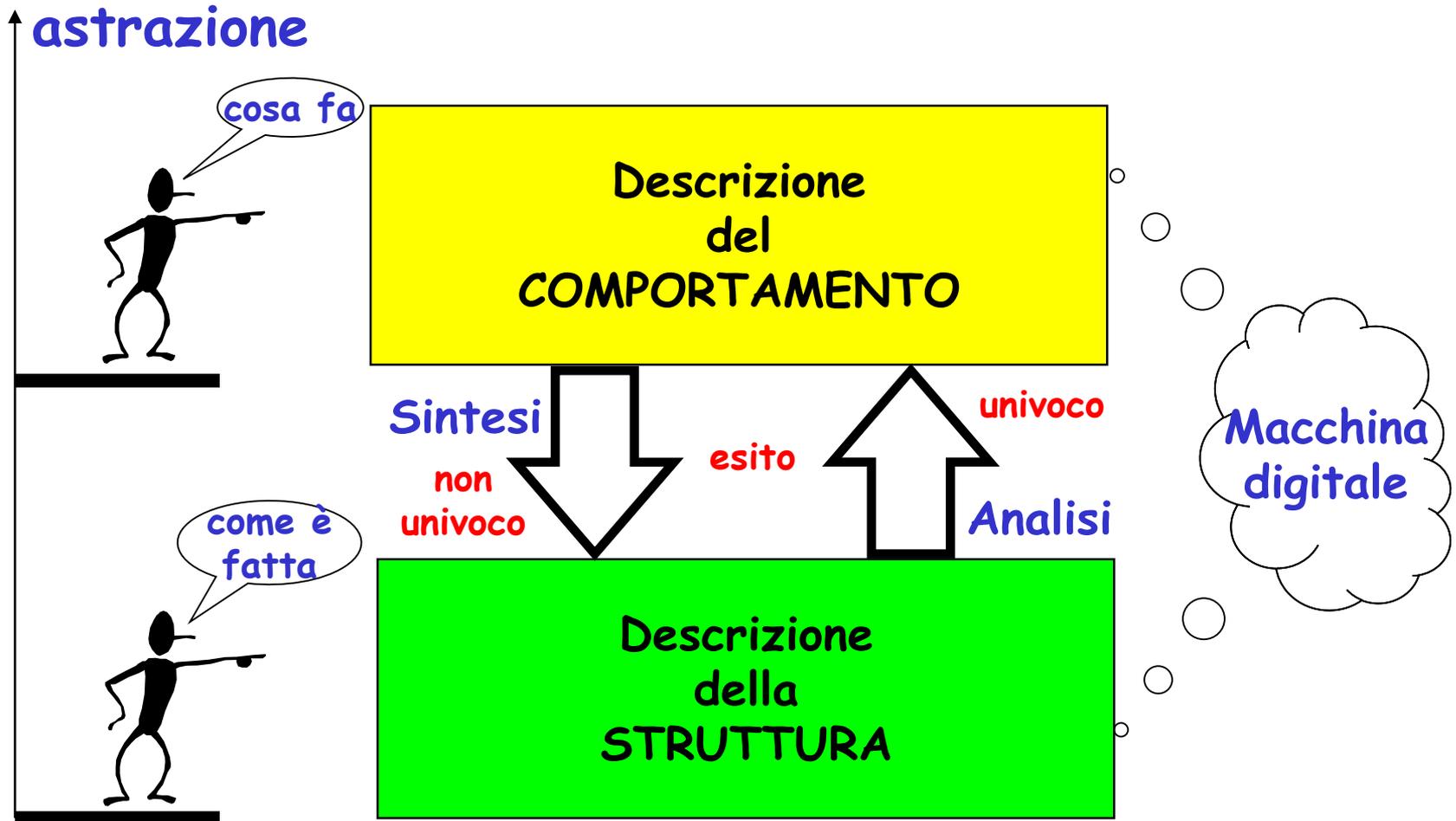
Una diffusissima macchina digitale: il calcolatore elettronico

- La «macchina universale» (general purpose) teorizzata da Charles Babbage nel 1833 viene poi realizzata con l'avvento della tecnologia elettronica da Alan Turing e John Von Neumann cento anni dopo (ENIAC, 1943)



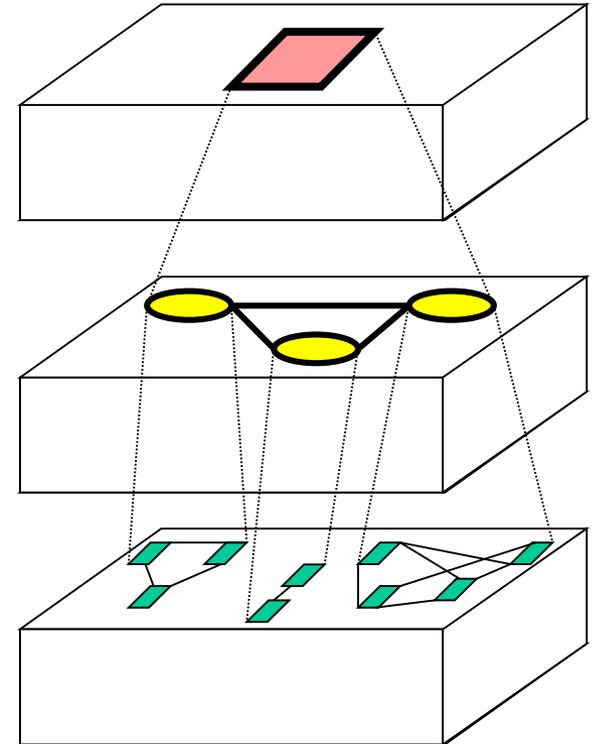
Analisi e sintesi

- La complessità di una macchina può essere definita sia a livello di comportamento, sia a livello di struttura
- Analisi e sintesi ci permettono di passare dalla **descrizione del comportamento** a quella della **struttura** (e viceversa)



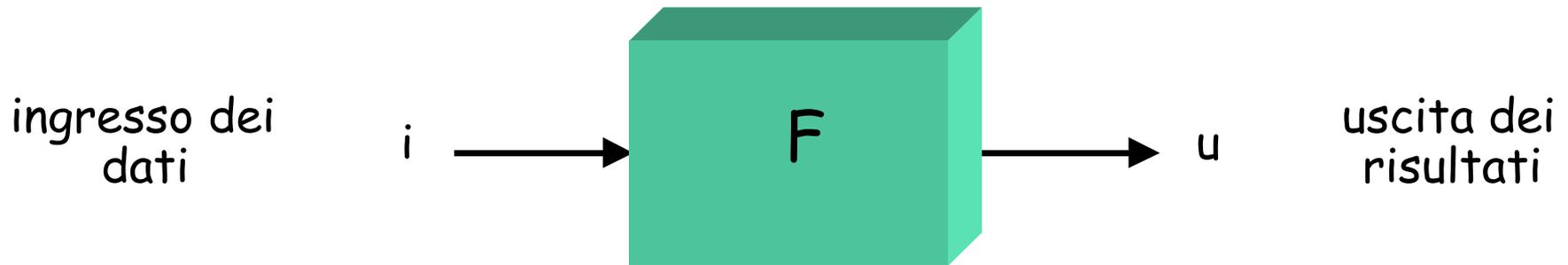
Livelli di astrazione

- In ogni ambito ingegneristico, la descrizione del comportamento e della struttura di un sistema complesso è inevitabilmente articolata su **più livelli** (approccio "*divide et impera*").
- Ogni livello individua **entità** (sottosistemi, unità, moduli, ..., componenti "primitivi") formate da componenti «astratti» la cui struttura è definita nel livello sottostante
- queste entità sono opportunamente cooperanti, contraddistinte da ben predefiniti ruoli, funzionalità, interfacce e protocolli di interazione con le altre entità operanti nello stesso livello o nei livelli adiacenti della gerarchia.
- Esplorando i livelli della gerarchia dall'alto verso il basso, **aumenta il numero di entità**, ma **diminuisce la complessità** di ciascuna di esse, dal punto di vista sia comportamentale che strutturale



Il modello black-box

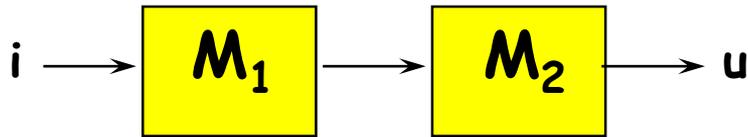
- Ogni componente è schematizzabile come una «scatola nera», caratterizzata da un insieme di segnali in ingresso e in uscita e da una funzione che ne determina la relazione



F: relazione di ingresso/uscita o di causa/effetto

Regole "elementari" di composizione

a) in serie

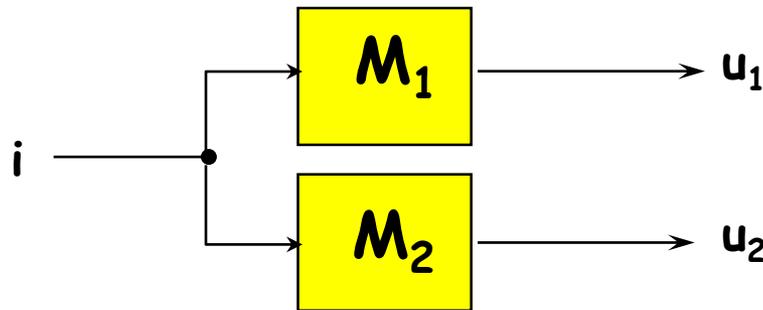


$$u = M_2(M_1(i))$$

Funzione composta

Deve operare prima il blocco a sinistra, poi quello a destra.

b) in parallelo

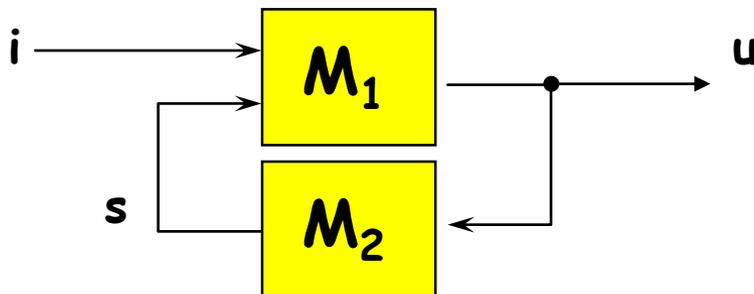


$$\begin{cases} u_1 = M_1(i) \\ u_2 = M_2(i) \end{cases}$$

Sistema di funzioni

I due blocchi operano contemporaneamente.

c) in retroazione



$$u = M_1(i, s)$$

$$s = M_2(u)$$

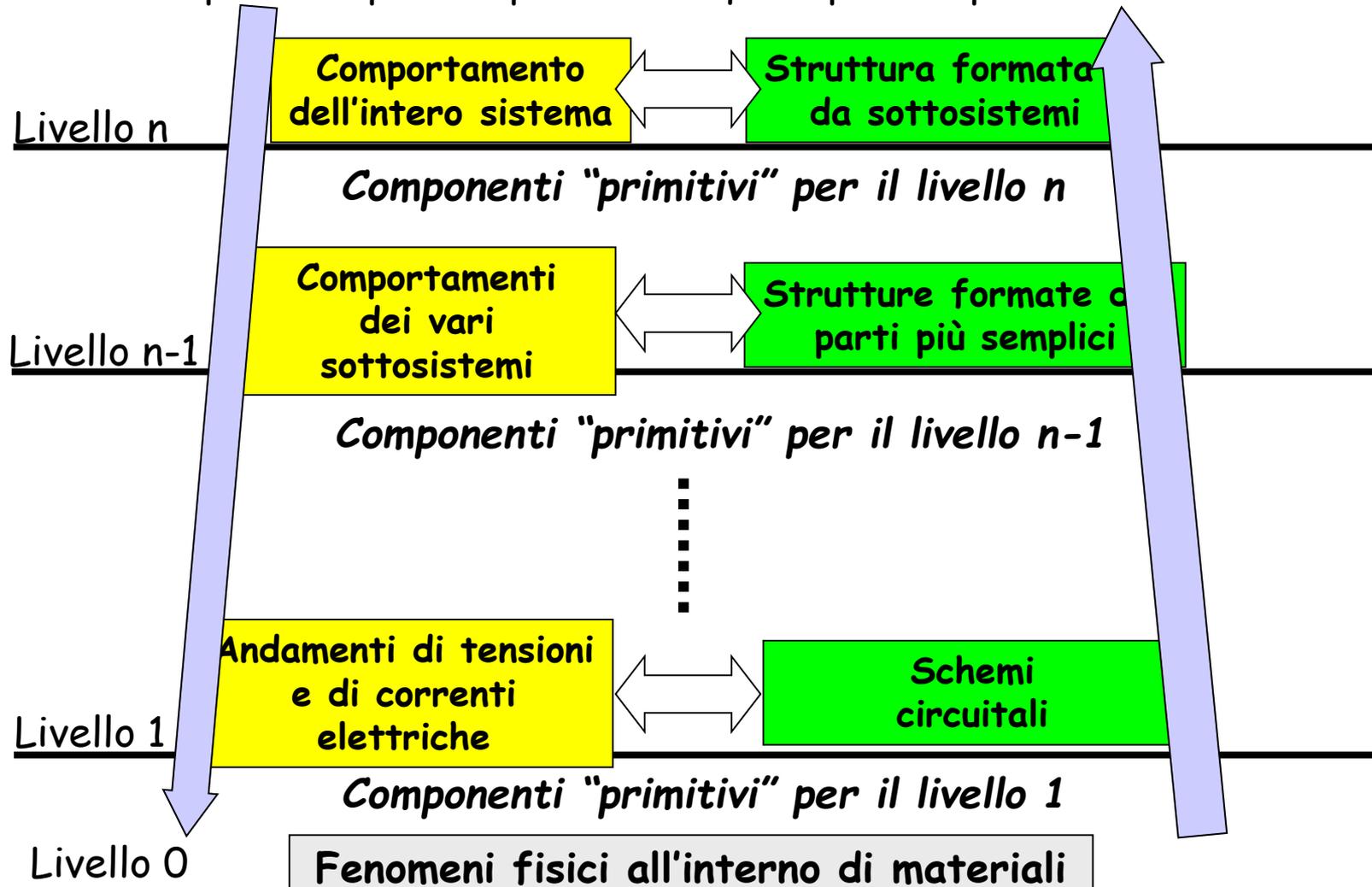
$$u = M_1(i, M_2(u^-))$$

Funzione ricorsiva

È necessario che l'anello completi un calcolo prima di avviarne uno nuovo.

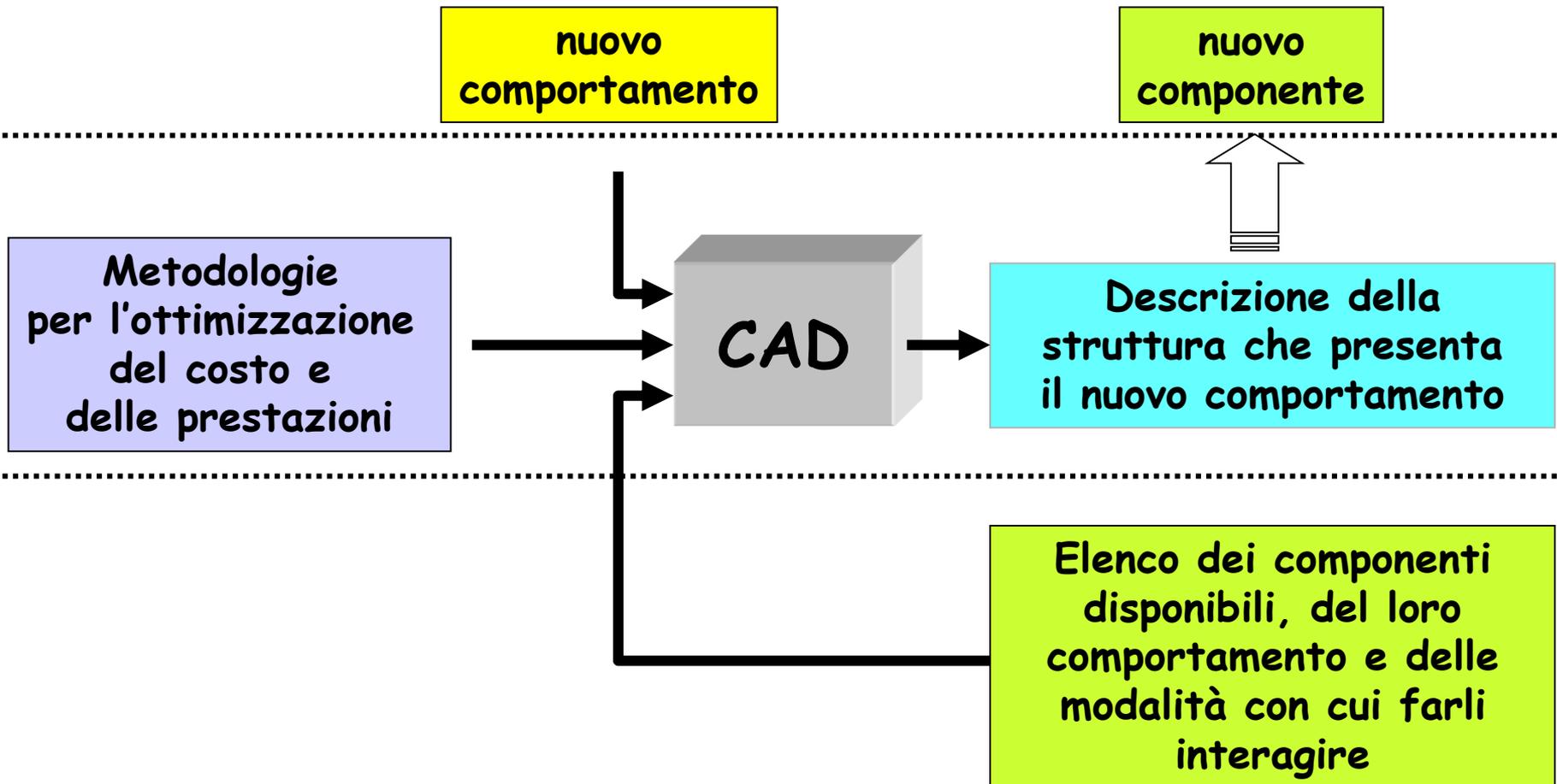
Progettazione top-down e bottom-up

- Progetto di una macchina complessa (*sintesi*):
 - prima **top-down**: definizione delle specifiche a livelli sempre più bassi
 - poi **bottom-up**: composizione graduale della macchina a partire da componenti più semplici verso quelli più complessi

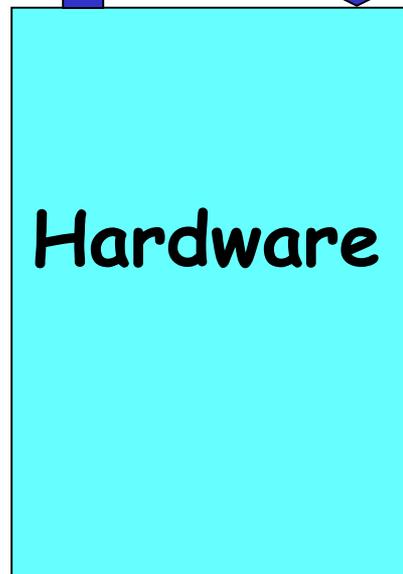


Il progetto, o sintesi, su un livello

- Assegnata un certo comportamento e una famiglia di componenti, la scelta non è univoca
- Il progettista deve scegliere valutando **costi e prestazioni**
- Computer Aided Design (**CAD**): aiuta il progettista in tale scelta



La macchina "programmabile" (general-purpose)



Linguaggio di Programmazione



Instruction Set

Livello
architettonico



Registri, Contatori, Selettori, Alu, ecc.

Livello
logico



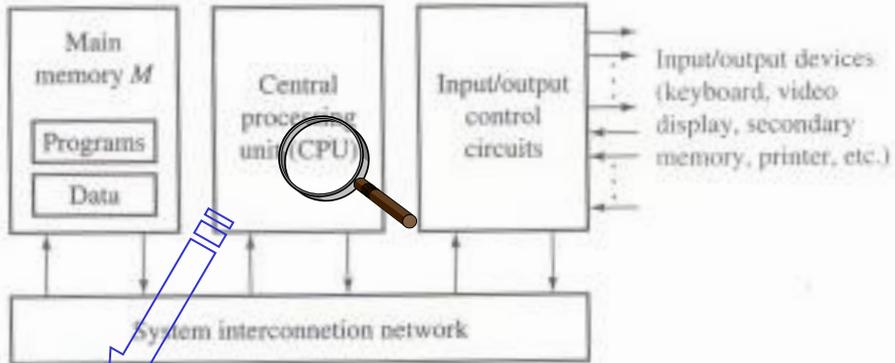
Famiglie e Librerie di Circuiti

Livello
fisico



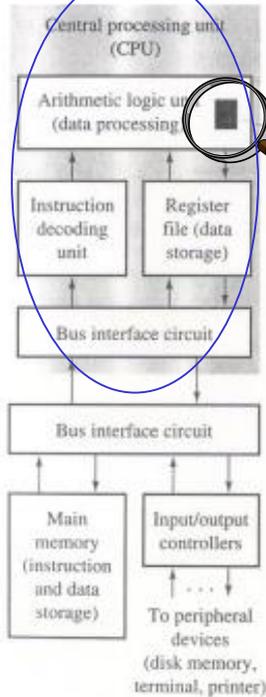
Interruttori elettronici

Fondamenti di Informatica



Livello architetturale

Calcolatori Elettronici



Livello logico

Reti Logiche

Livello fisico

Microelettronica

Elettronica Digitale

Rete Logica:

modello della macchina digitale che consente

- di astrarre dalla tecnologia
- di dettagliare l'immagine architeturale

RAM, Registro,
Contatore, ALU,
Decoder, Multiplexer

processore,
memoria, I/O

Funzioni,
variabili,
espressioni

Livello architeturale

Adattabilità, velocità,
capacità, sicurezza,
espressività

Livello logico

Descrizione formale,
composizione,
decomposizione

segnali e
circuiti

Livello fisico

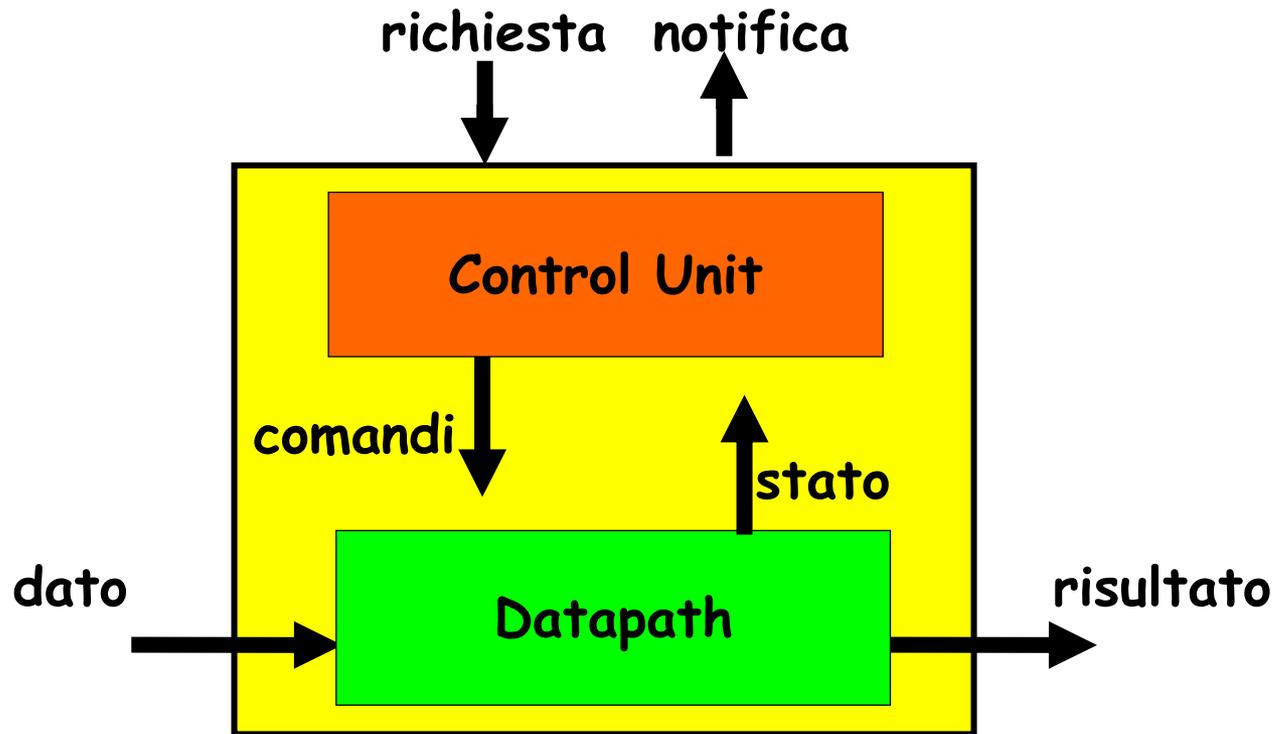
Affidabilità, velocità,
ingombro, consumo,
costo

Argomenti da affrontare per impiegare il modello:

- Modalità di rappresentazione, elaborazione e trasferimento dell'informazione
- Metodi per la definizione formale delle specifiche
- Metodologie di progetto

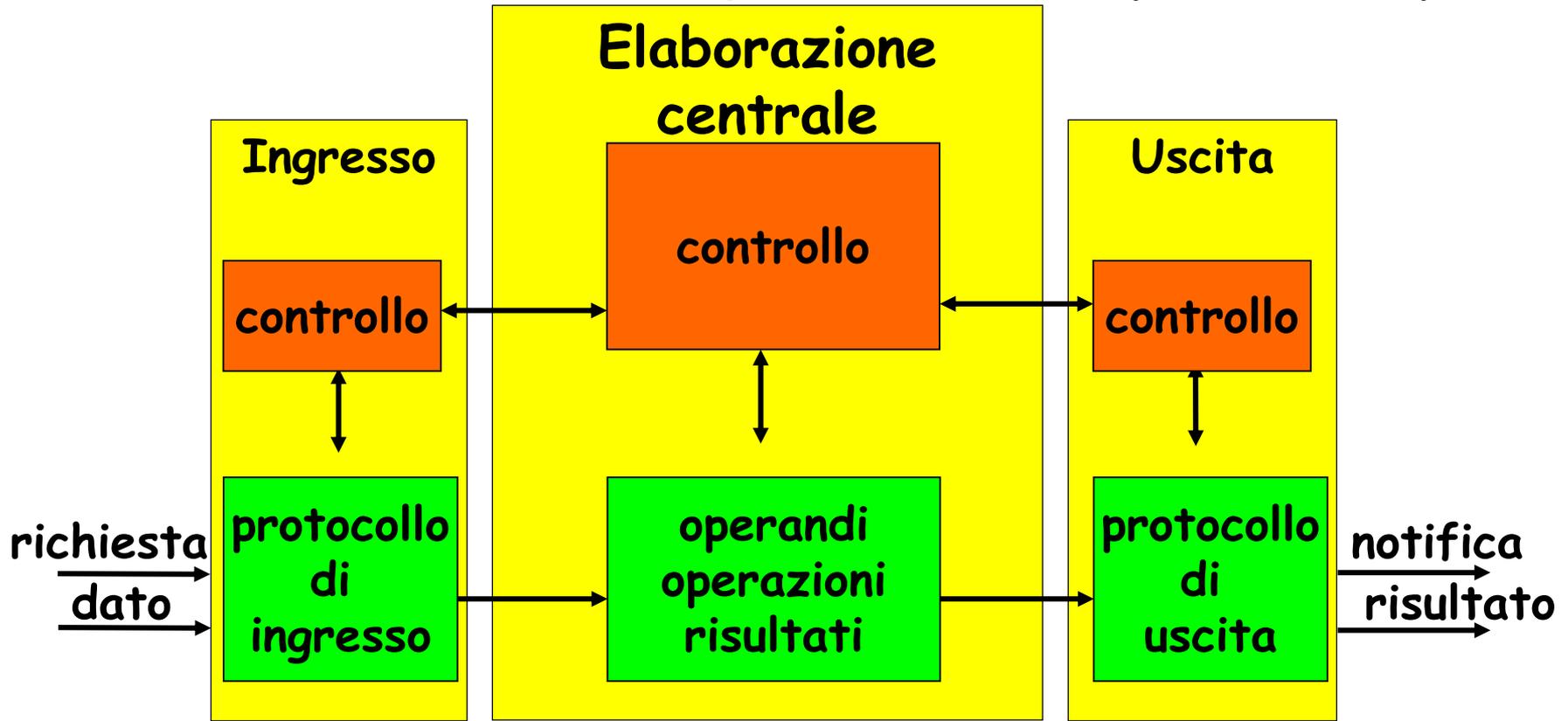
1.1 Unità funzionali

Il Controllo ed il Percorso dei dati



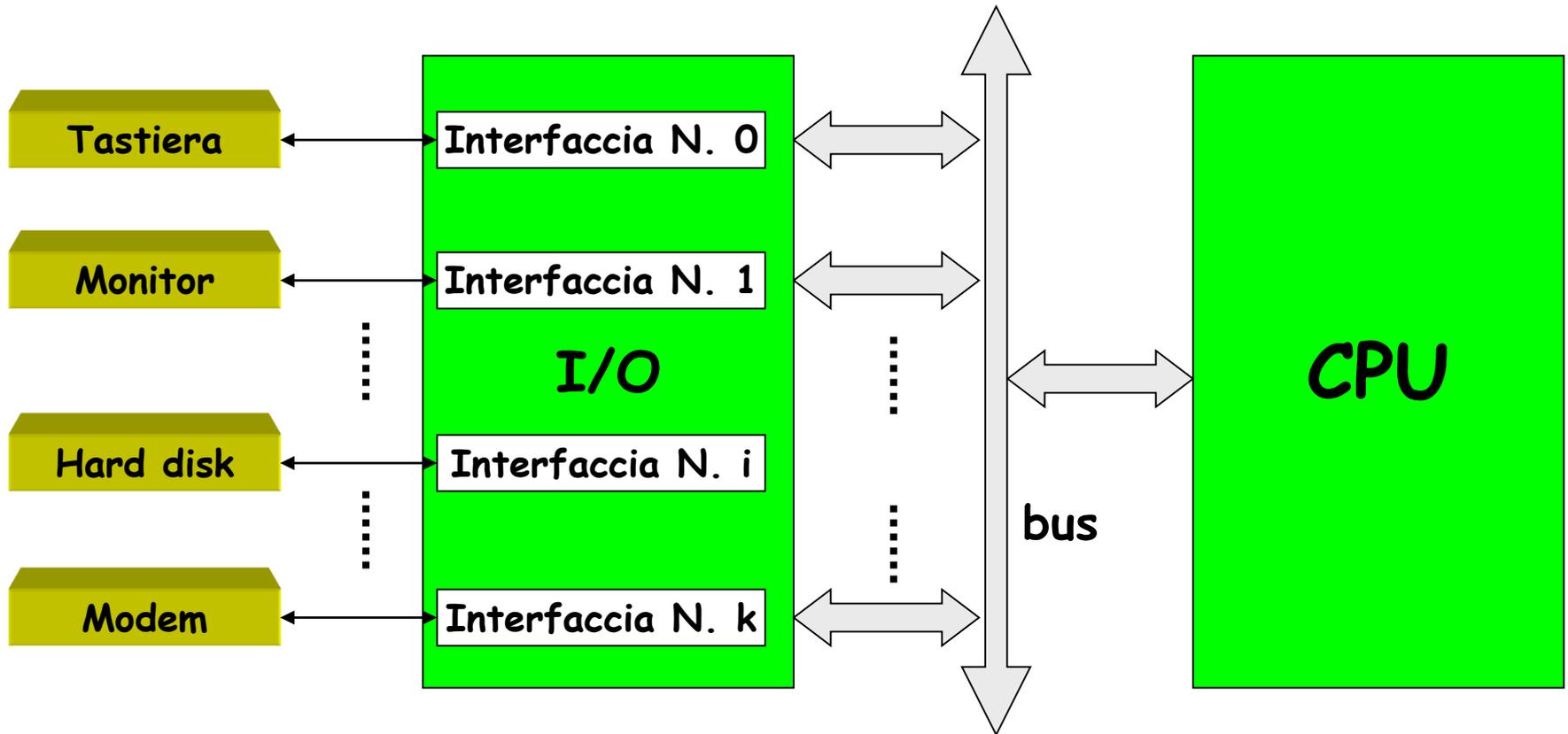
- L'hardware di un calcolatore è formato da due blocchi
 - **Control Unit**: fornisce in ogni istante i comandi operativi associati a ciascun passo dell'algoritmo in esecuzione
 - **Datapath**: esegue le operazioni sui dati in ingresso mediante i comandi forniti dalla Control Unit
- Struttura **flexibile**: si possono ottenere comportamenti diversi utilizzando il medesimo hardware (cambiando cioè dati e richieste, relativi al *software*)

Central Processing Unit e Input/Output



- Un calcolatore ha necessità di scambiare informazioni con l'esterno
- Occorrono specifici protocolli di ingresso e uscita da utilizzare nelle interfacce di ingresso (input) e uscita (output) che gestiscano sincronizzazione dei dati, traduzione, etc..
- Vantaggio: l'elaborazione centrale non si deve occupare dello scambio di informazioni con l'esterno

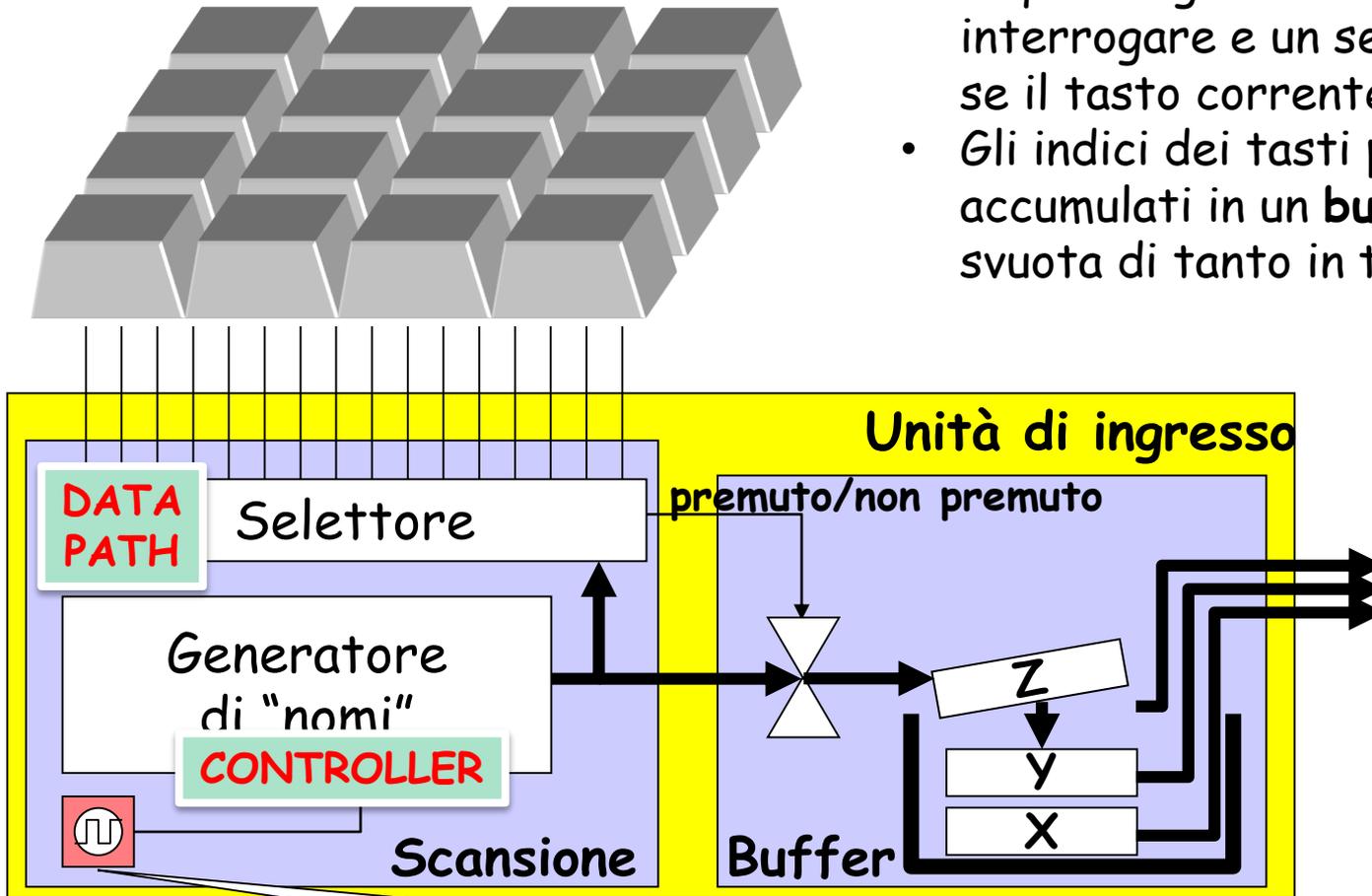
I/O, bus, interfacce e dispositivi



- Ciascun dispositivo di I/O necessita di una specifica interfaccia per comunicare con la CPU
- Tali interfacce vengono riassunte all'interno del blocco di I/O
- Analizzeremo ora due interfacce comuni (tastiera e video)

L'interfaccia della tastiera

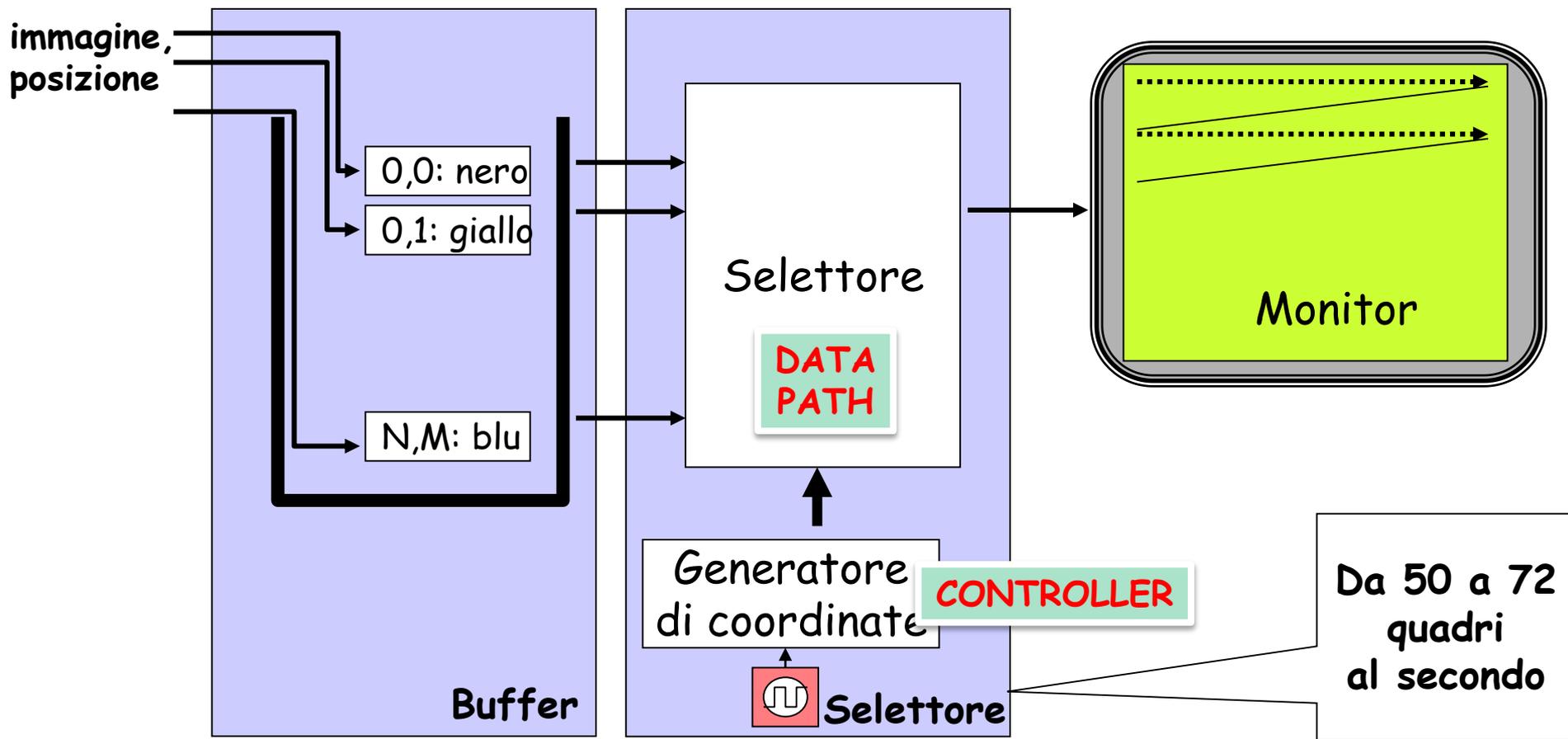
- **Unità di scansione (a polling)**: ha un generatore di nomi che genera in sequenza gli indici dei tasti da interrogare e un selettore che indica se il tasto corrente è premuto o meno
- Gli indici dei tasti premuti vengono accumulati in un **buffer** (che la CPU svuota di tanto in tanto)



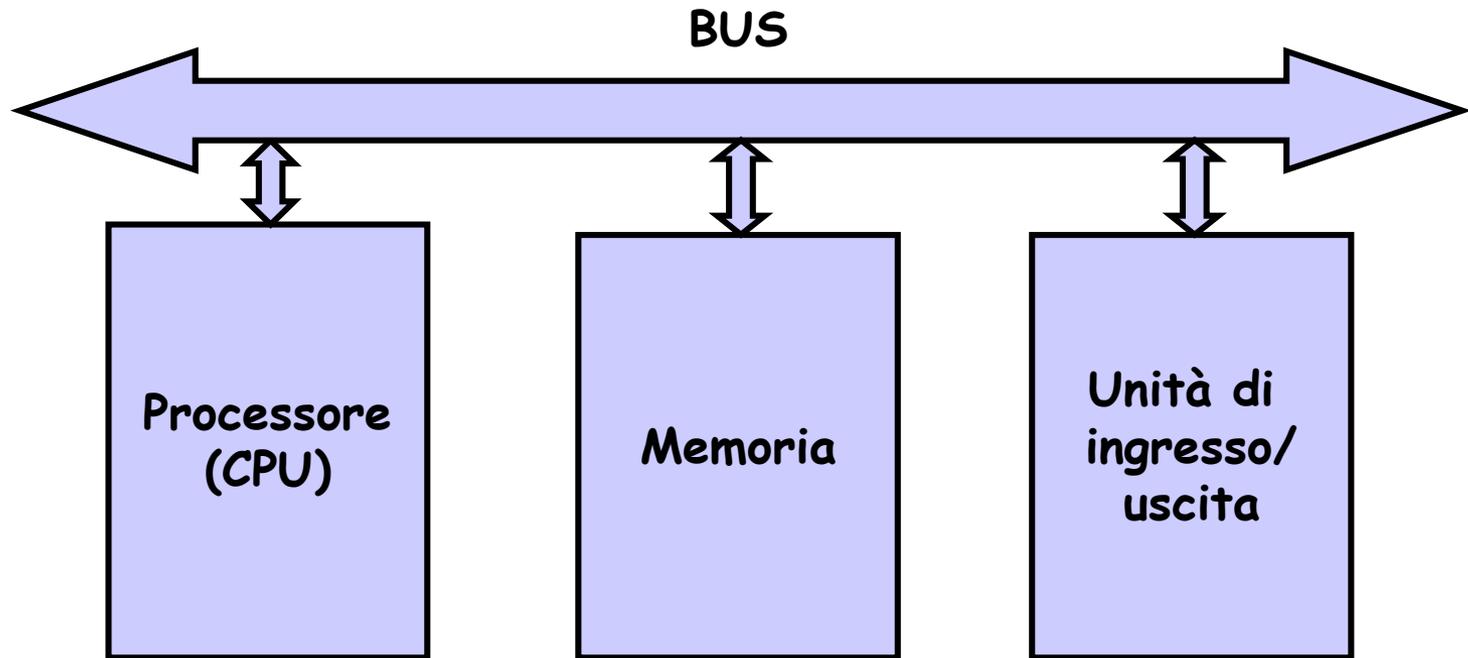
Da 10 a 30 interrogazioni al secondo

Interfaccia Video

- **Buffer:** memorizza ciascuna schermata in ingresso
- **Selettore:** seleziona un dato alla volta nel buffer e lo invia al monitor, ove un pennello elettronico applica il colore corrente nella posizione corrente
- **Generatore di coordinate:** indica di volta in volta quale dato prelevare dal buffer



Architettura di un calcolatore elettronico

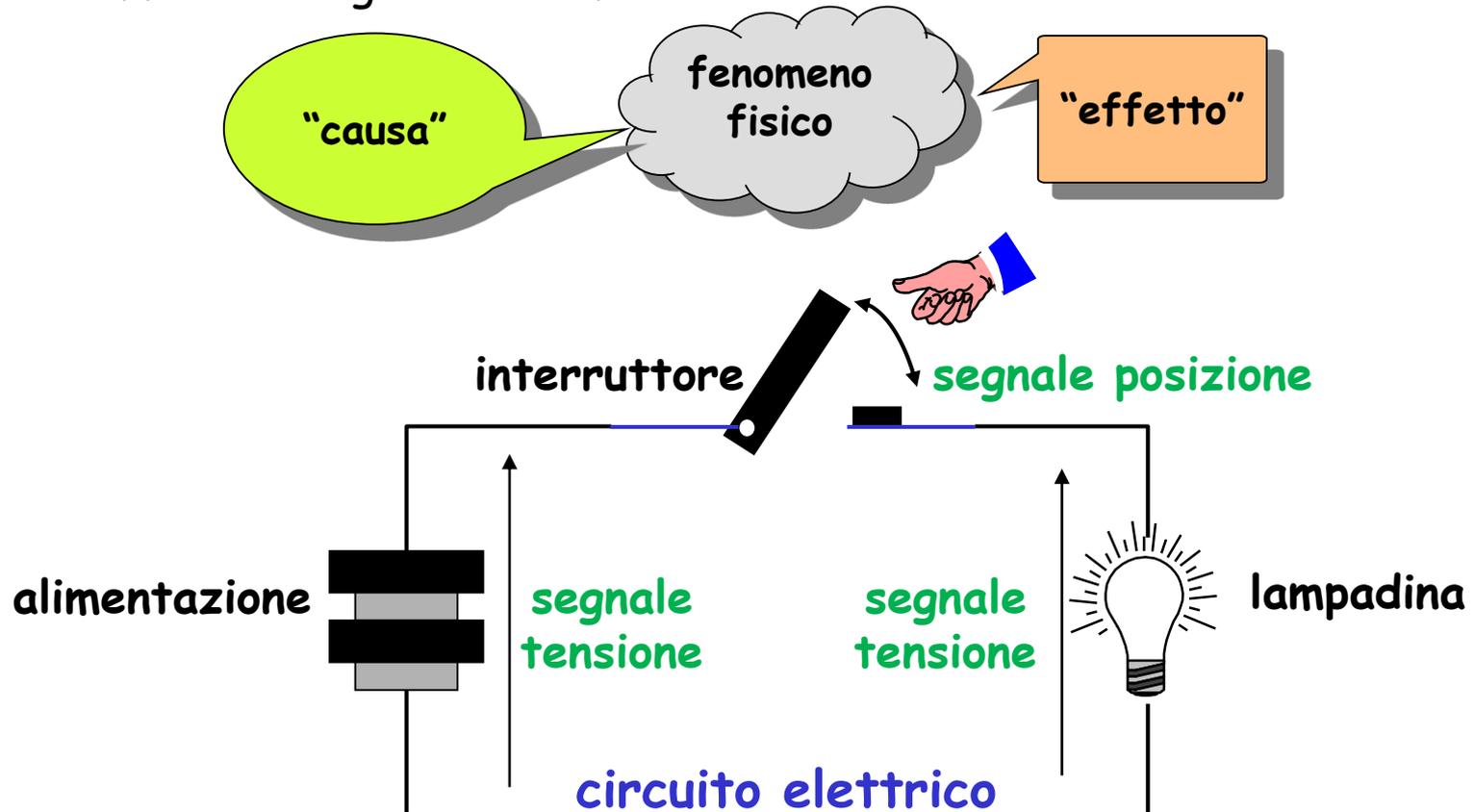


- In un'ottica più generale, l'elaborazione centrale sta all'interno di un unico blocco (**processore**) collegato ai dispositivi di I/O mediante un unico canale di comunicazioni ad alta velocità (**bus**) (*architettura di Von Neumann*)
- La **memoria** rappresenta un ulteriore blocco contenente le istruzioni e i dati, anch'esso interconnesso agli altri mediante il bus

1.2 Macchine digitali: dal livello fisico al livello logico

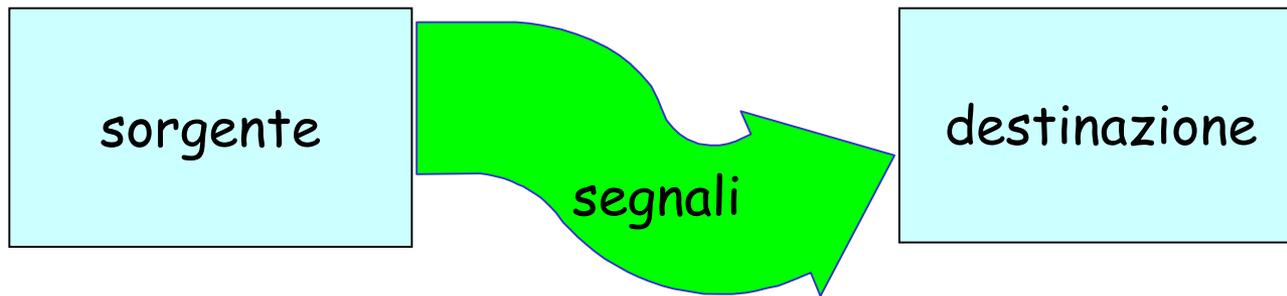
Il livello fisico

- A livello fisico si prendono in considerazione relazioni di causa/effetto tra grandezze fisiche (**segnali**) che caratterizzano i fenomeni fisici
- Ad esempio: in un circuito elettrico, il segnale tensione ai capi di una lampadina dipende dal segnale tensione di alimentazione e dal segnale posizione dell'interruttore
- Problema del **progetto**: coordinare lo svolgimento di innumerevoli relazioni di causa/effetto tra grandezze fisiche



Analogico vs. Digitale

I circuiti elettronici che formano il livello fisico di una macchina digitale coordinano il loro funzionamento scambiandosi informazioni veicolate da "segnali".



Segnale: grandezza fisica variabile nel tempo, il cui **andamento** o **forma d'onda** identifica l'**informazione** che la sorgente intende inviare alla destinazione.

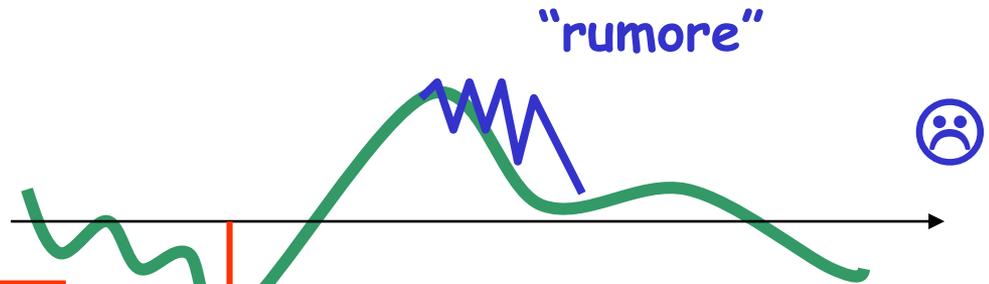
Segnale analogico: ogni variazione della grandezza fisica modifica l'informazione trasportata. Dispone (teoricamente) di infiniti valori.

Segnale digitale: solo a "significative" variazioni della grandezza fisica corrisponde una modifica dell'informazione trasportata. Dispone di un numero finito di valori

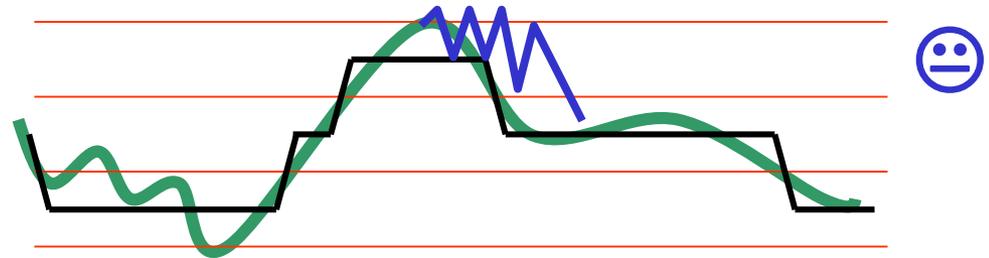
Forma d'onda di un segnale

- Il segnale analogico

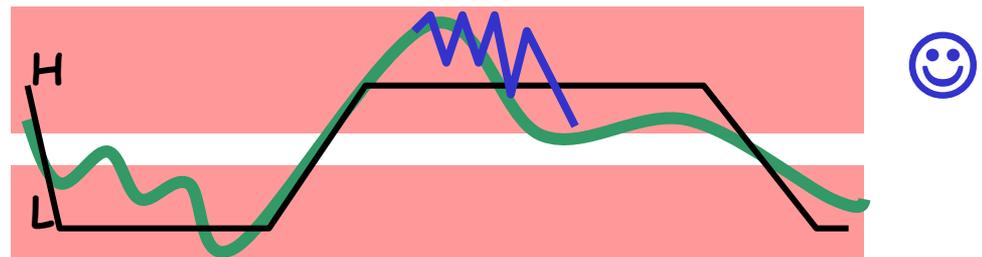
$s(t) \equiv$ informazione



- Il segnale digitale



- Il segnale binario



Segnali analogici vs. segnali digitali

PROBLEMA: comunicare il valore N di un numero intero ($0 \leq N \leq 999$).

IPOTESI: si dispone di una tensione elettrica V variabile nell'intervallo 0-10 volt, di cui si è in grado di generare (lato sorgente) e misurare (lato destinazione) il valore con la precisione del centesimo di volt (0.01 V).

SOLUZIONI

Segnale analogico: posto $V=N$, la comunicazione richiede **una sola unità di tempo**, ma un "rumore" di entità pari a **0,01 volt** altera il valore trasferito.

Es.: $N=250$

$V=2.50$

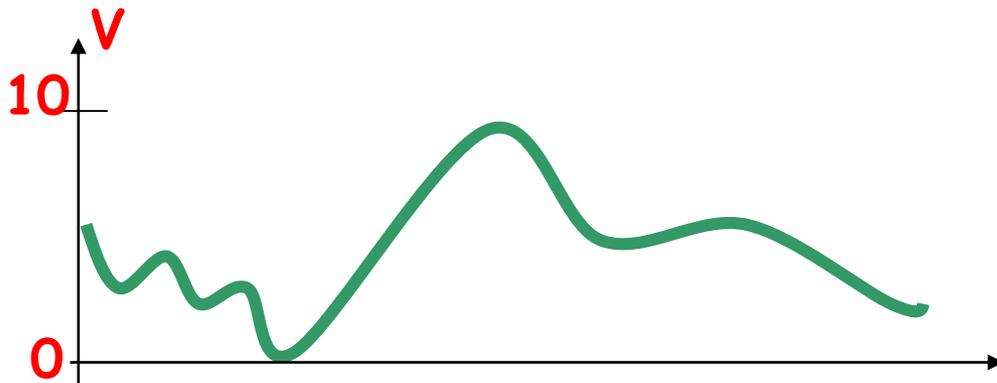
Rumore=0.01V

$V=2.51$
-> $N=251$



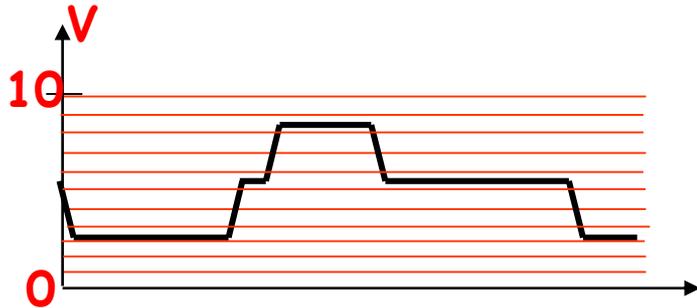
Rumore=1V

$V=3.50$
-> $N=350$

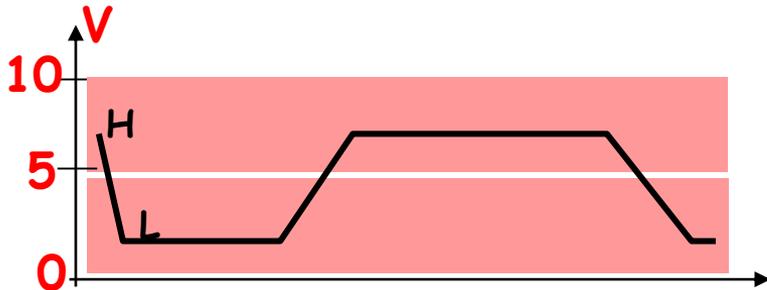


Segnali analogici vs. segnali digitali

Segnale digitale: suddiviso l'intervallo di variabilità del segnale V in 10 fasce da 1 volt, la comunicazione richiede **tre unità di tempo**, una per ciascuna cifra decimale; l'immunità al rumore è pari a **0,5 volt**.



Segnale binario: suddiviso l'intervallo di variabilità del segnale V in 2 fasce da 5 volt, la comunicazione richiede **dieci unità di tempo**; l'immunità al rumore è pari a **2,5 volt**.



Es.: $N=250$

$$V=2.50 - 5.50 - 0.50$$

$$v_1=2.5 \text{ V}$$

$$v_2=7.5 \text{ V}$$

$$v_1 - v_1 - v_2 - v_2 - v_2 - v_2 - v_2 - v_1 - v_2 - v_1$$

Rumore=0.01V

$$V=2.51 - 5.51 - 0.51 \rightarrow N=250$$

$$v_1=2.51 \text{ V}$$

$$v_2=7.51 \text{ V}$$

$$\rightarrow N=250$$



Rumore=1V

$$V=3.50 - 6.50 - 1.50 \rightarrow N=361$$

$$v_1=3.5 \text{ V}$$

$$v_2=8.5 \text{ V}$$

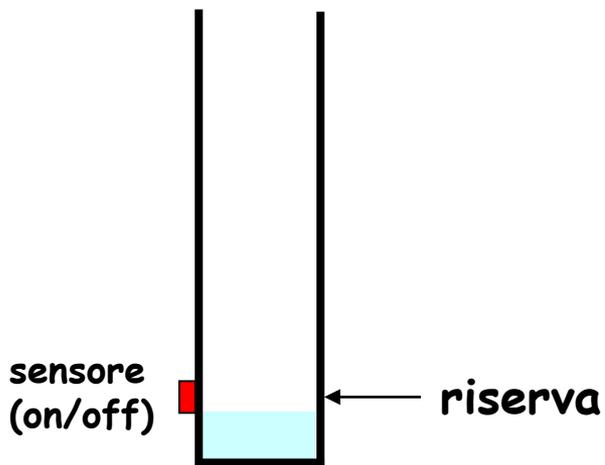
$$\rightarrow N=250$$



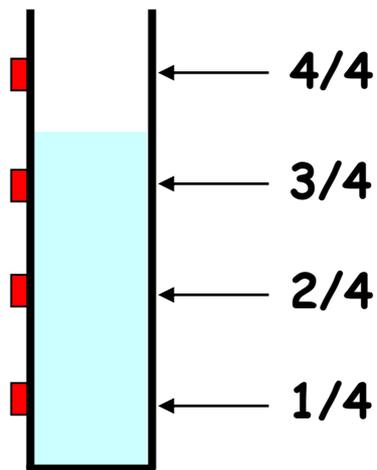
Segnali analogici vs. segnali binari

PROBLEMA: evidenziare il livello del carburante in un'automobile

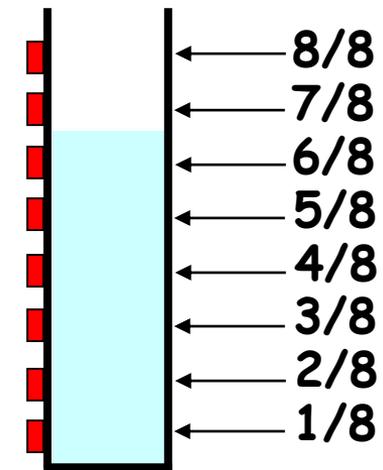
SOLUZIONI: 1 segnale analogico, oppure, a parità di contenuto informativo, una molteplicità di segnali binari



1 segnale



4 segnali



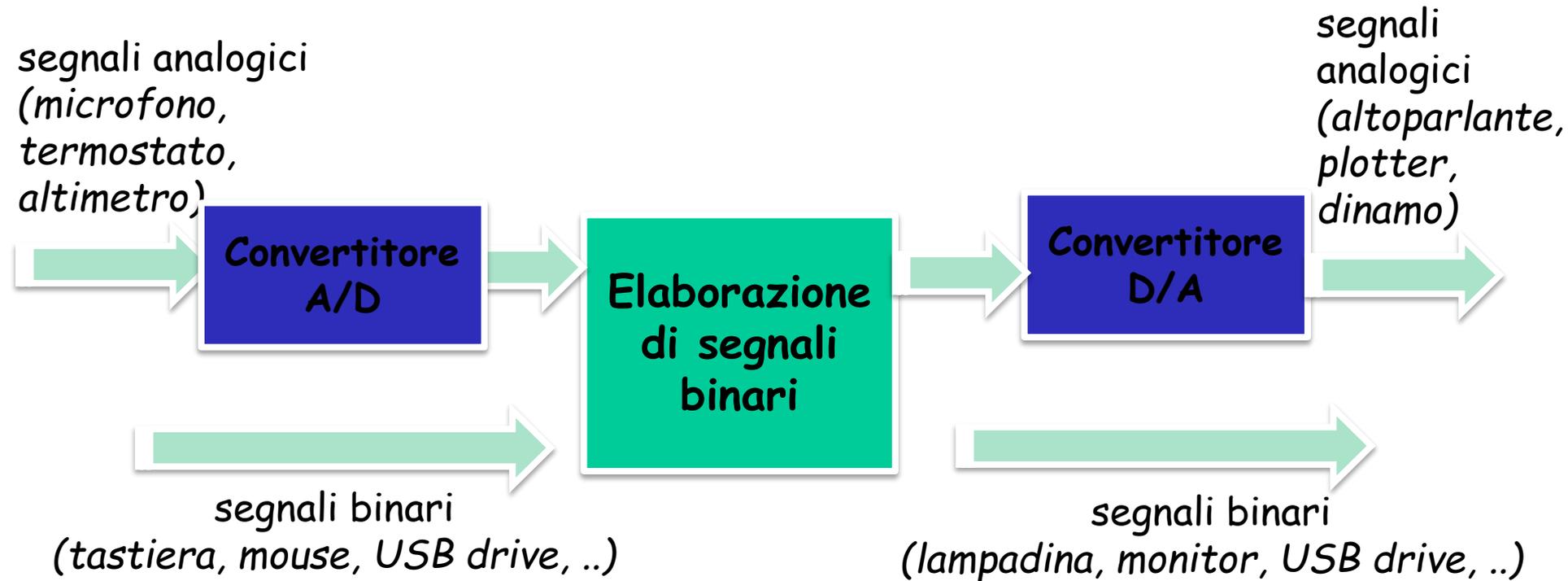
8 segnali

...



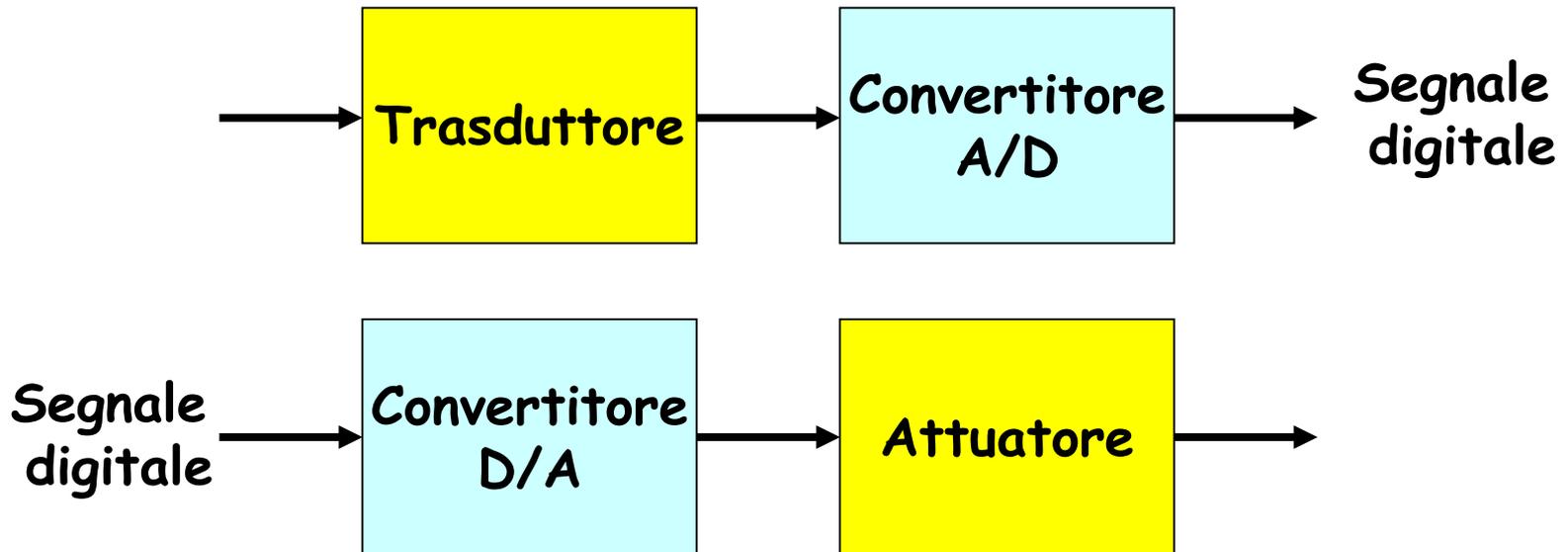
Conversione A/D e D/A

- Per scambiare informazioni con il mondo esterno attraverso segnali analogici una macchina digitale si avvale di convertitori *analog-to-digital* (A/D) e *digital-to-analog* (D/A)



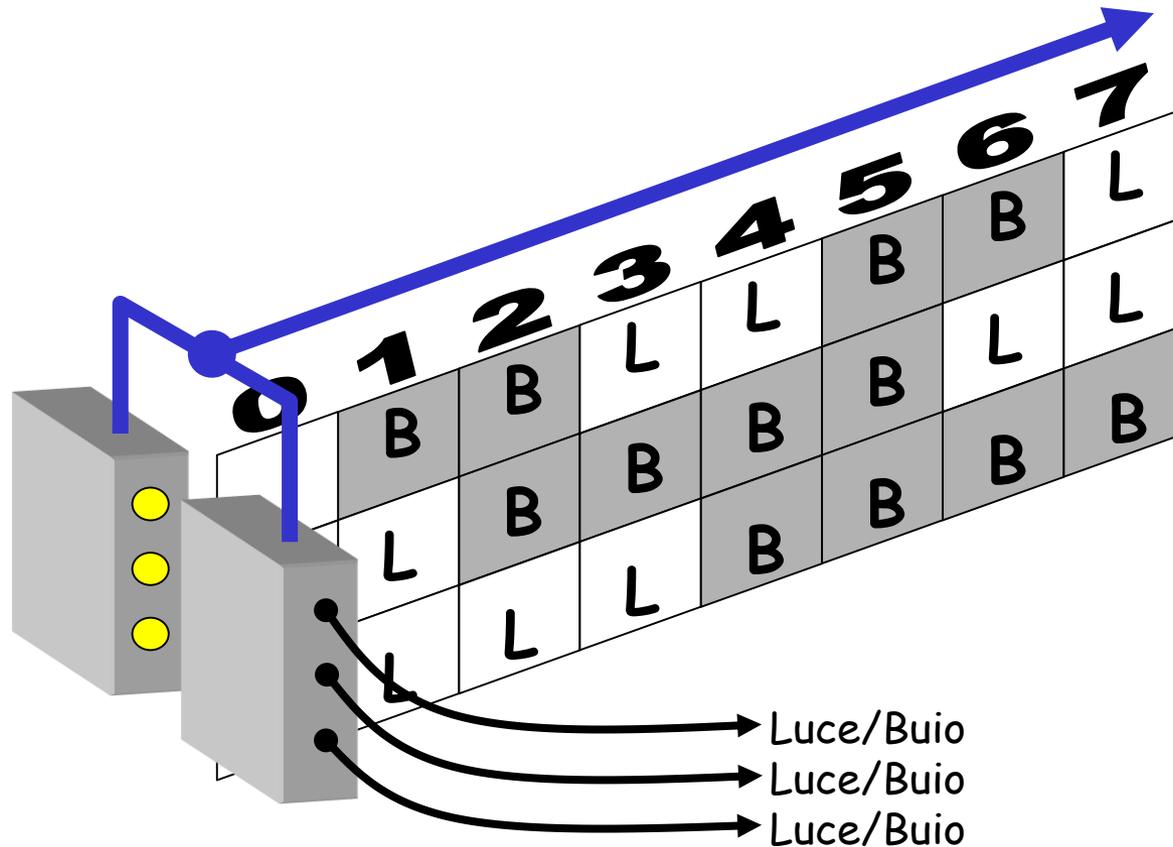
Trasduttori, Convertitori, Attuatori

- I convertitori A/D e D/A sono spesso utilizzati per convertire segnali di grandezze diverse da quella elettrica (meccanica, acustica,...) mediante trasduttori e attuatori



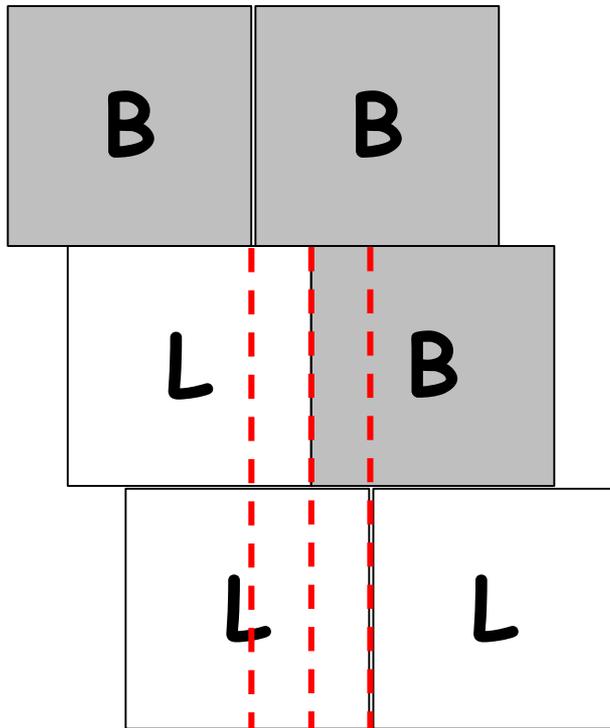
Esempio: conversione A/D

- Una slitta portautensili trasporta 3 **fotodiodi** e 3 **fotocellule**
- In ciascuna delle 8 posizioni, si hanno configurazioni differenti di luce e buio per via della diversa trasparenza del materiale interposto
- Si converte in questo la posizione della slitta da segnale analogico a digitale (*discretizzata* in 8 posizioni)
- Ogni posizione viene così descritta tramite **3 segnali binari**
- Posizioni consecutive differiscono di un solo bit (come mai?)

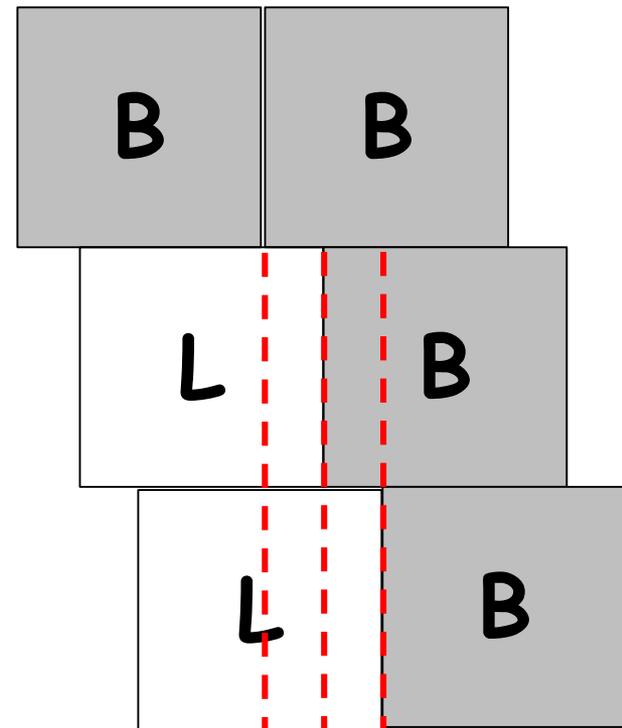


Esempio: conversione A/D (2)

- L'allineamento meccanico perfetto è in pratica impossibile
- Per evitare letture scorrette nei punti di disallineamento è opportuno utilizzare configurazioni relative a posizioni consecutive che differiscono di un solo valore



B B B B B B
L L L B B B
L L L L L L



B B B B B B
L L L B B B
L L L L B B

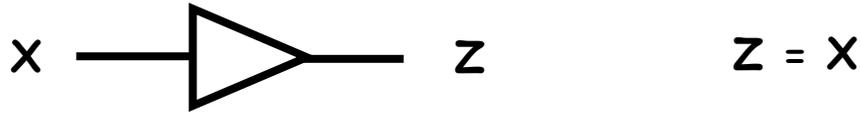


Lettura:

1.3

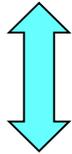
Gate elementari

Il gate "Buffer"

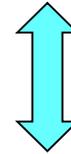


X	Z
0	0
1	1

↑
astrazione



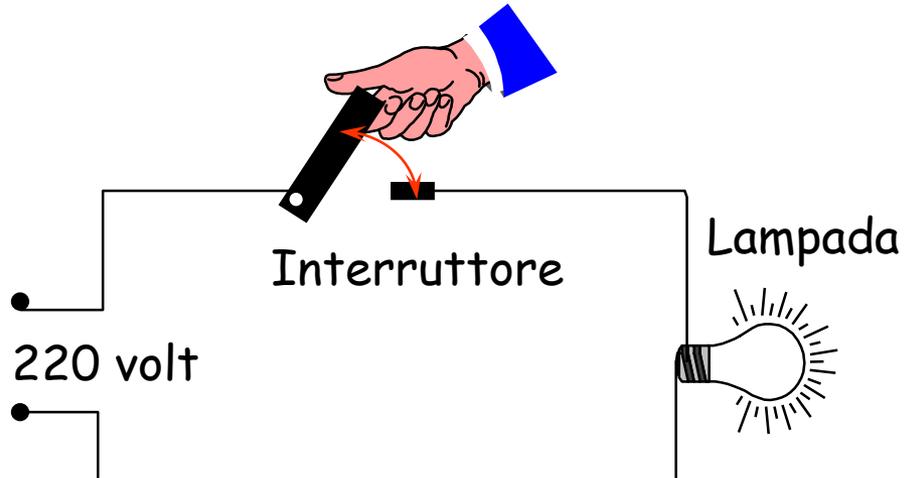
Livello logico



Livello fisico

Struttura

Comportamento



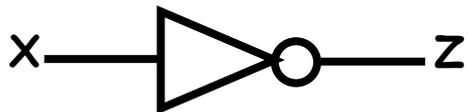
posizione del contatto	lampadina
alta	spenta
bassa	accesa

ANALISI della relazione causa/effetto

- la causa è la posizione del contatto
- l'effetto è l'emissione o meno di luce



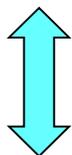
Il gate "NOT"



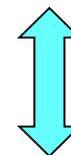
$$Z = X'$$

X	Z
0	1
1	0

astrazione



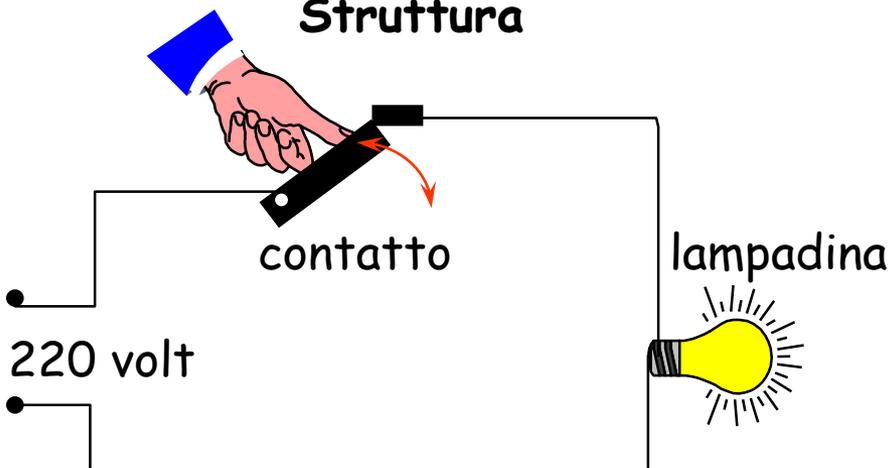
Livello logico



Livello fisico

Struttura

Comportamento



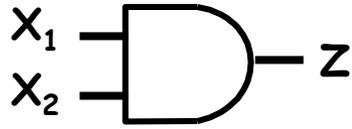
posizione del contatto	lampadina
alta	accesa
bassa	spenta

ANALISI

- la causa è la posizione del contatto
- l'effetto è l'emissione o meno di luce



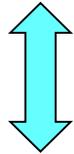
Il gate "AND"



$$Z = X_1 \cdot X_2$$

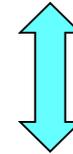
X_1	X_2	Z
0	0	0
0	1	0
1	0	0
1	1	1

astrazione



Livello logico

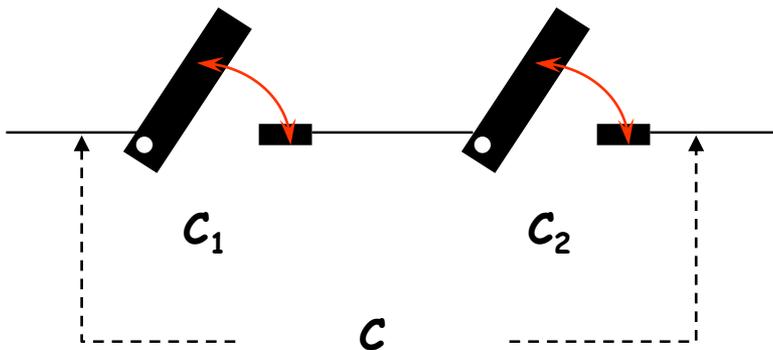
Livello fisico



Struttura

Comportamento

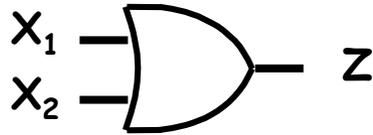
Contatti in serie



C_1	C_2	C
aperto	aperto	aperto
aperto	chiuso	aperto
chiuso	aperto	aperto
chiuso	chiuso	chiuso

C è chiuso se sono chiusi C_1 e C_2

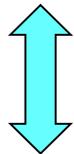
Il gate "OR"



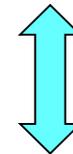
$$Z = X_1 + X_2$$

X_1	X_2	Z
0	0	0
0	1	1
1	0	1
1	1	1

astrazione



Livello logico

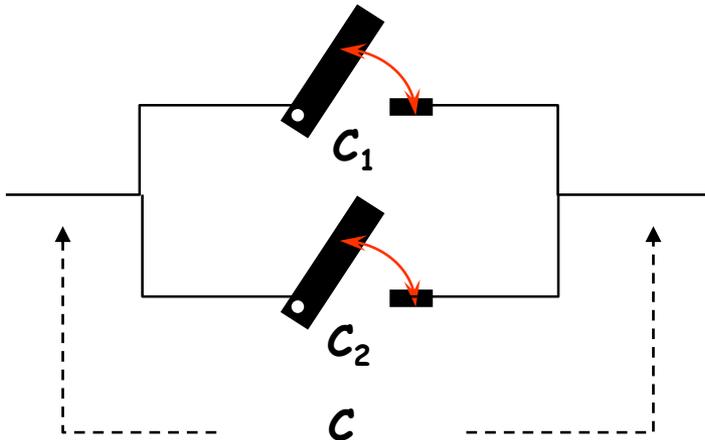


Livello fisico

Struttura

Comportamento

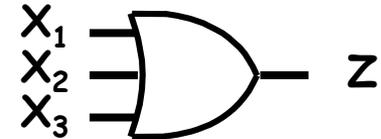
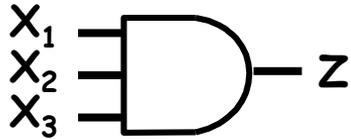
Contatti in parallelo



C_1	C_2	C
aperto	aperto	aperto
aperto	chiuso	chiuso
chiuso	aperto	chiuso
chiuso	chiuso	chiuso

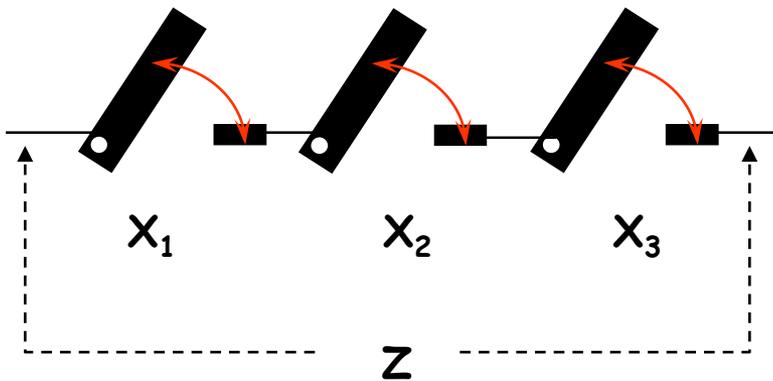
C è chiuso se è chiuso $\circ C_1 \circ C_2$

AND e OR con più di due ingressi

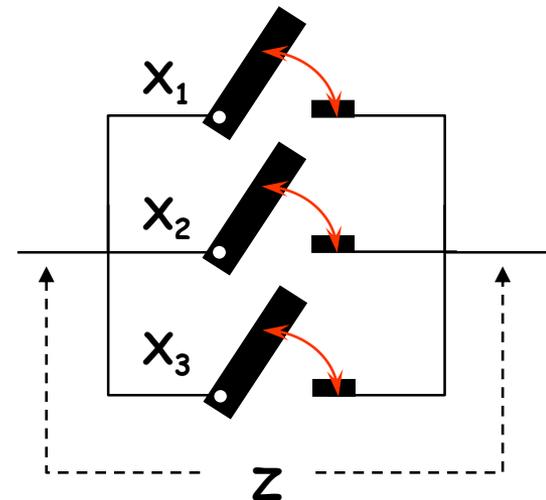


N. B. - Il numero dei segnali di ingresso di un gate è detto *fan-in*

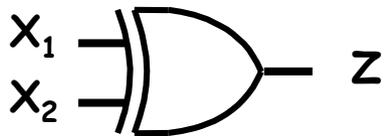
Contatti in serie



Contatti in parallelo



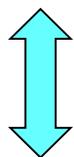
Il gate "EX-OR"



$$Z = X_1 \oplus X_2$$

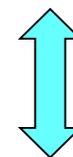
X_1	X_2	Z
0	0	0
0	1	1
1	0	1
1	1	0

astrazione



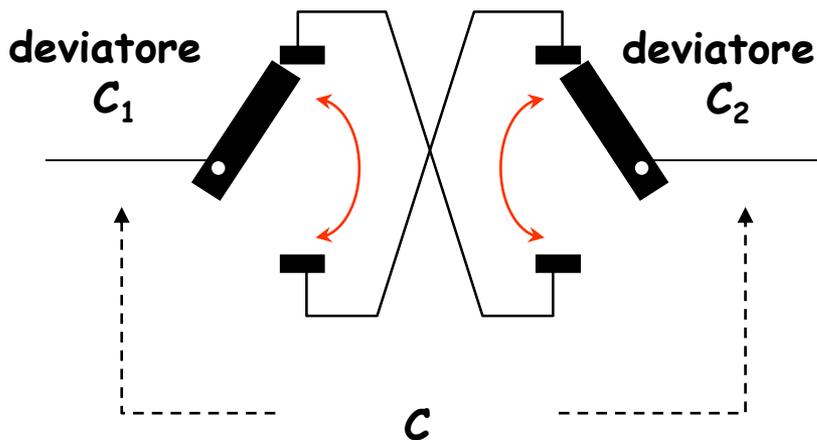
Livello logico

Livello fisico



Struttura

Comportamento

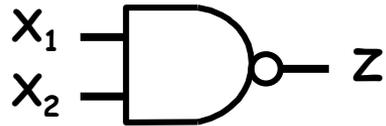


C_1	C_2	C
aperto	aperto	aperto
aperto	chiuso	chiuso
chiuso	aperto	chiuso
chiuso	chiuso	aperto

C è chiuso se è chiuso C_1 o C_2 ,
ma non entrambi

Altri gate elettronici elementari

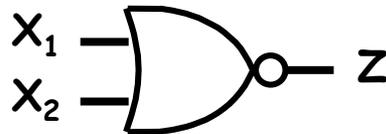
NAND



$$Z = (X_1 \cdot X_2)'$$

X_1	X_2	Z
0	0	1
0	1	1
1	0	1
1	1	0

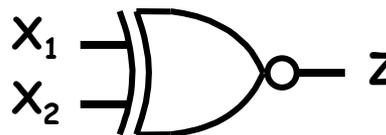
NOR



$$Z = (X_1 + X_2)'$$

X_1	X_2	Z
0	0	1
0	1	0
1	0	0
1	1	0

EQUIVALENCE



$$Z = (X_1 \oplus X_2)'$$

X_1	X_2	Z
0	0	1
0	1	0
1	0	0
1	1	1

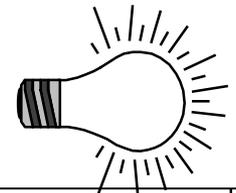
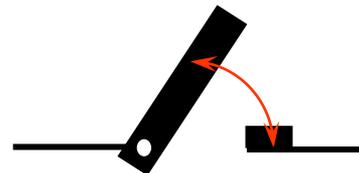
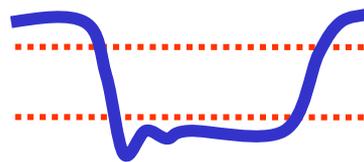
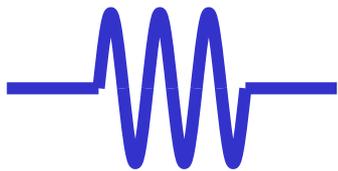
Variabili binarie

- Per descrivere a livello logico un segnale binario si utilizza una variabile, detta variabile binaria o **bit** (binary digit), che può assumere i due soli valori «0» e «1»

Bit: variabile x tale che $x \in B : \{0,1\}$

logica positiva o negativa

Segnali binari: {Presente, Assente} {Alta, Bassa}
{Aperto, Chiuso} {Accesa, Spenta} ecc.



x	corrente	x
1	assente	0
0	presente	1

x	tensione	x
1	bassa	0
0	alta	1

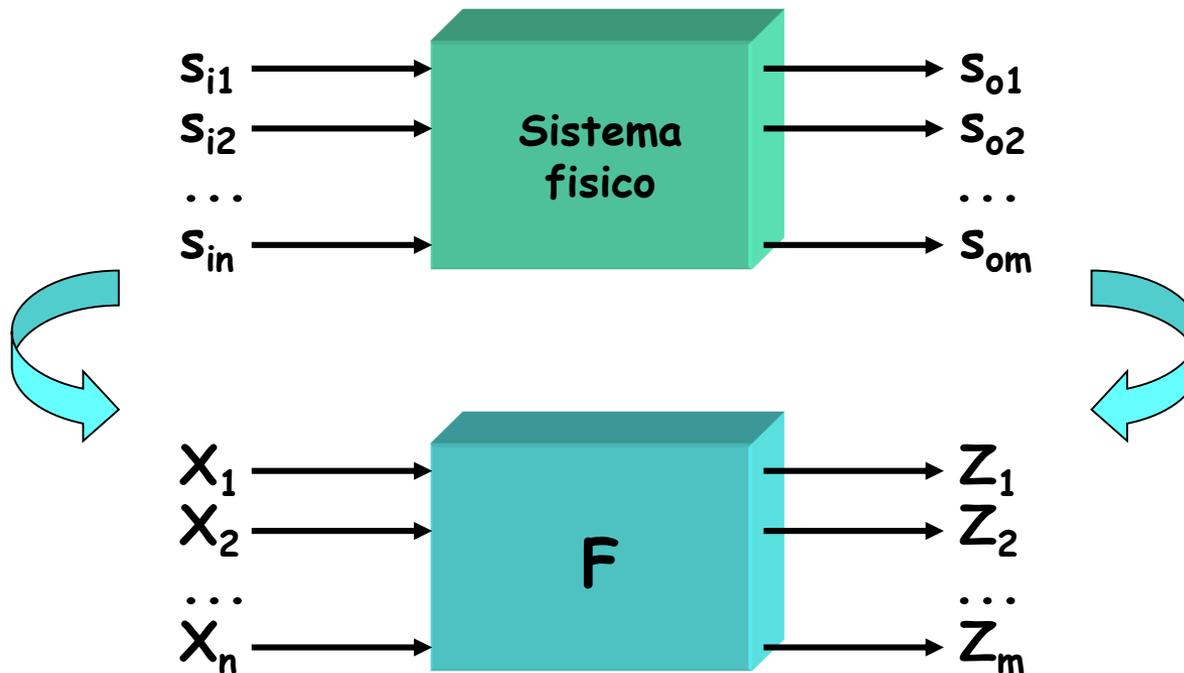
x	contatto	x
1	aperto	0
0	chiuso	1

x	lampadina	x
1	spenta	0
0	accesa	1

logica negativa logica positiva

Il modello black-box «reloaded»

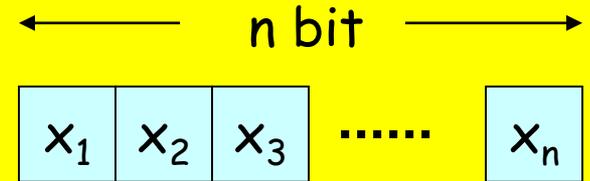
- Il sistema fisico è una «scatola nera» con segnali in ingresso e in uscita
- I segnali in ingresso e uscita ad un sistema fisico rappresentato da una macchina digitale sono variabili di ingresso e di uscita **binarie**



F: relazione di ingresso/uscita o di causa/effetto

Configurazioni binarie

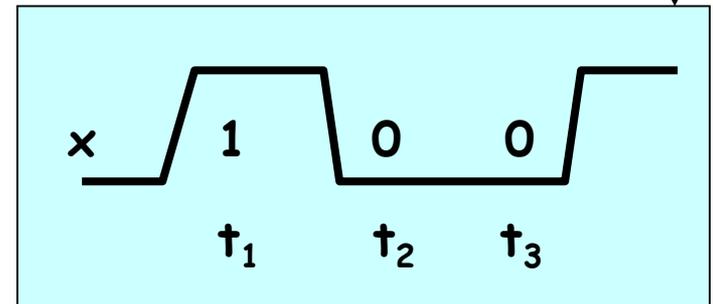
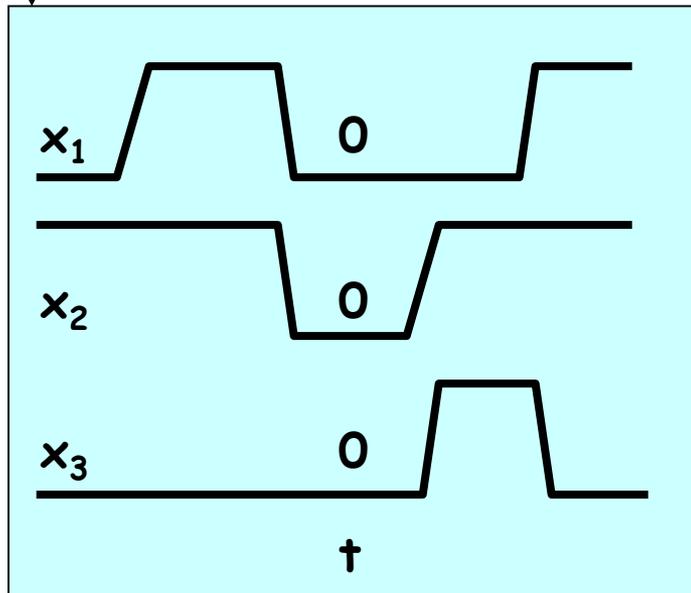
Configurazione binaria di n bit:
stringa di n simboli $\in \{0,1\}$.



- Le distinte configurazioni binarie di n bit sono 2^n
- Una configurazione di n bit può rappresentare:
 - i valori di n segnali binari in un certo istante;
 - i valori di un segnale binario in n istanti.

n=3:

x_1	x_2	x_3
0	0	0
1	0	0
0	1	0
0	0	1
1	1	0
1	0	1
0	1	1
1	1	1



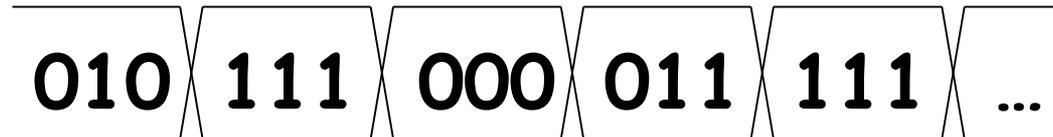
notazione alternativa:

$x_1 x_2 x_3$ 010 111 000 011 111 ...

Diagrammi ad occhio

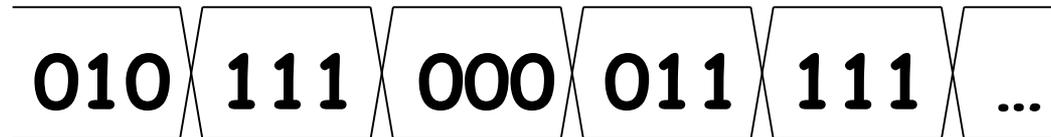
- Notazione utilizzata per rappresentare la sequenza di valori assunti da una (o più) variabili binarie al trascorrere del tempo
- Es.: descrizione dell'andamento di 3 variabili binarie:

$x_1 x_2 x_3$



- Due diagrammi ad occhio consentono di descrivere una relazione di causa/effetto

$x_1 x_2 x_3$



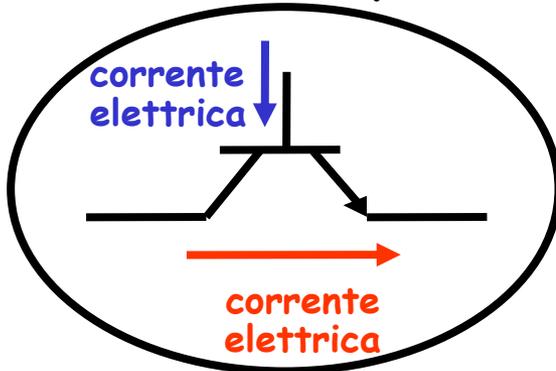
$z_1 z_2$



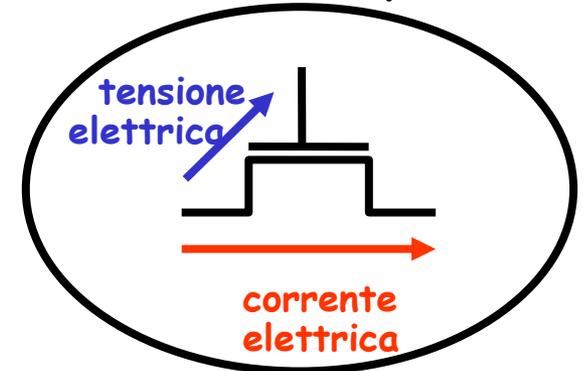
"Interruttori" elettronici

- Il transistor realizza l'**interruttore elettronico**: il passaggio della corrente tra due morsetti è regolato dal terzo morsetto variando corrente o tensione
- La relazione causa-effetto è caratterizzata da un *ritardo*

transistore bipolare

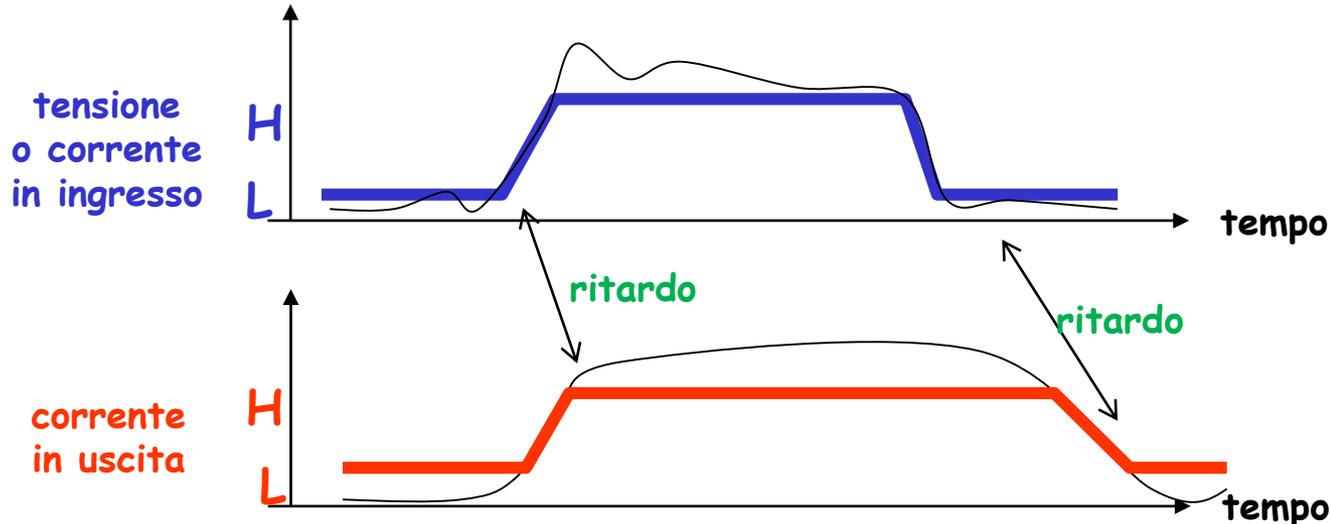


transistore unipolare



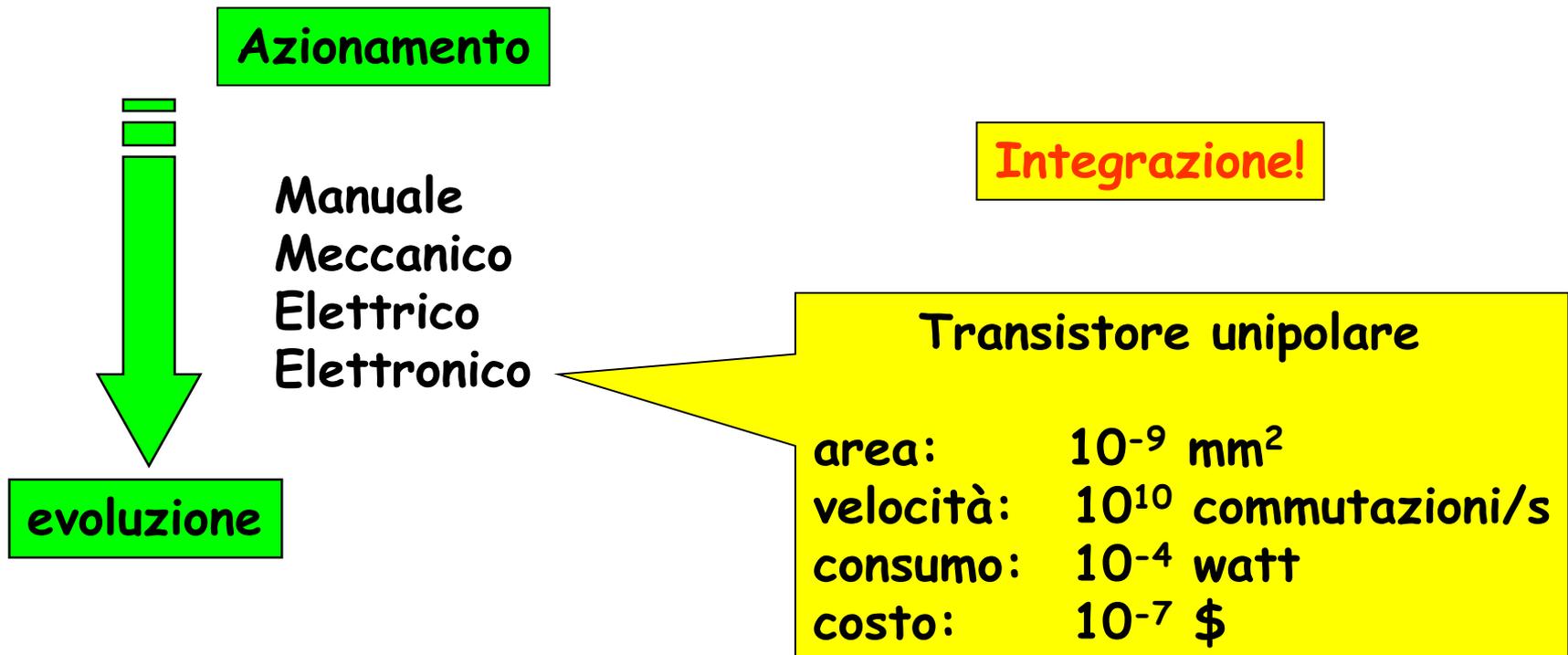
causa
valore "alto" / "basso"

effetto
corrente SI / NO



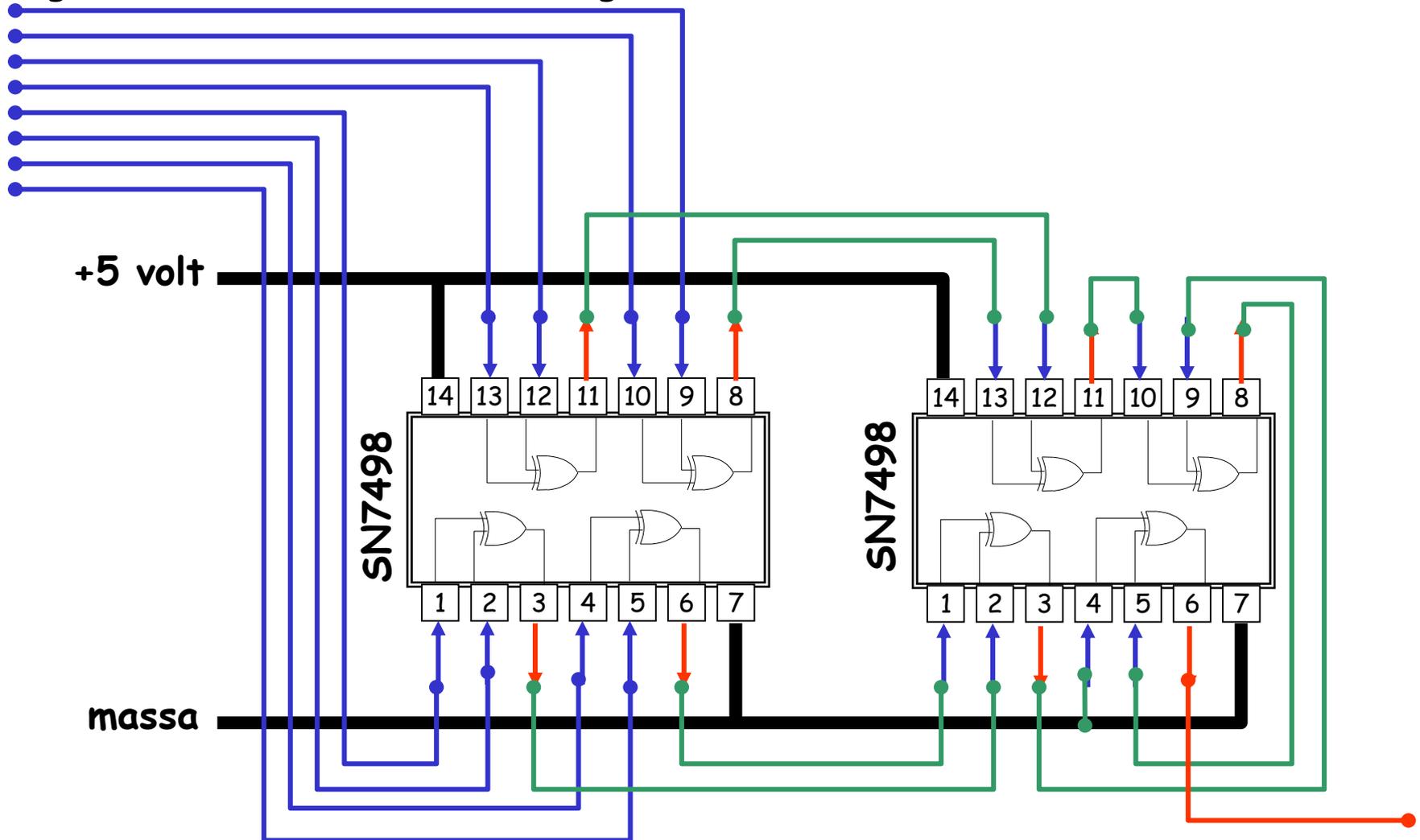
Tecnologia e prestazioni

- Vantaggi evidenti dovuti all'introduzione del transistor: riduzione dei costi, dell'ingombro, dei consumi, aumento della velocità di commutazione
- Possibilità di realizzare componenti integrati (**chip**)
- Ciascun componente integrato è descritto dal **data sheet** (descrizione della struttura e del comportamento, norme d'impiego, ..)

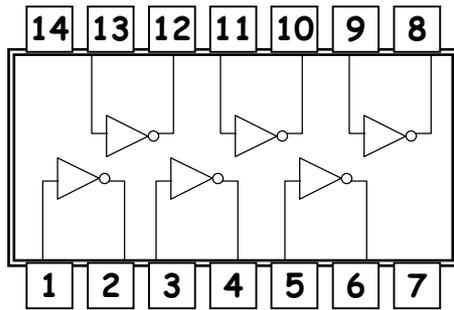


Tecnologia elettronica: collegamenti

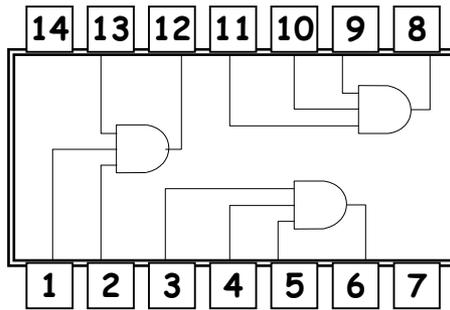
- Famiglie di gate: all'interno di un unico chip si trovano più gate dello stesso tipo
- Segnali collegabili ai piedini: ingresso, uscita e alimentazione (GND, VCC)
- Diverse realizzazioni: Transistor-Transistor Logic (TTL), Resistor-Transistor Logic (RTL), Diode-Transistor Logic (DTL)



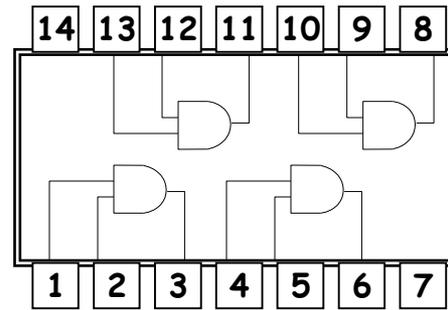
Famiglie di gate (TTL SSI -1968/74)



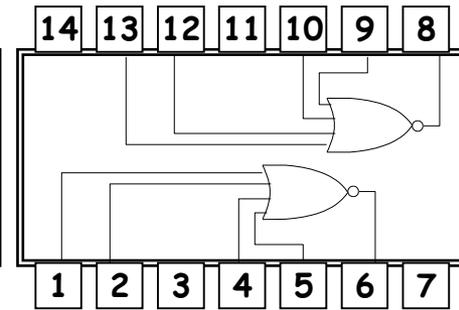
SN7404



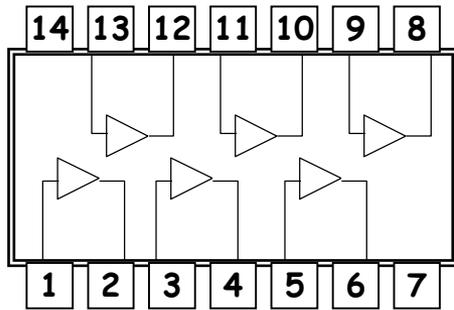
SN7411



SN7408



SN7423

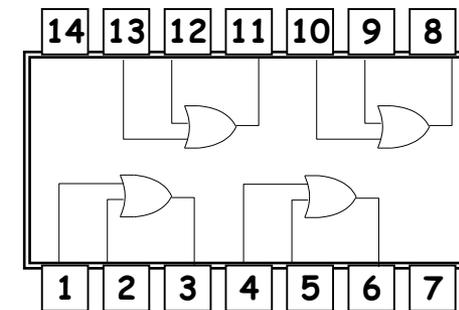


SN7407

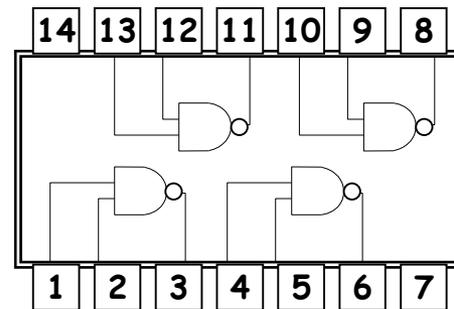
SSI: small scale integration

Il **data sheet** ne descrive le principali caratteristiche:

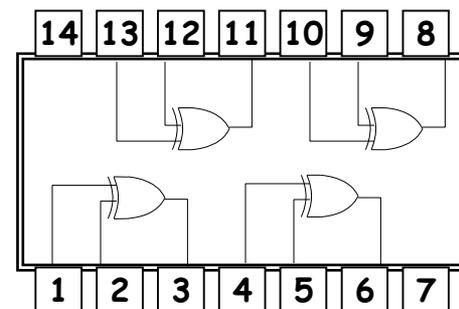
- alimentazione e consumo
- struttura logica interna
- segnali ai piedini
- fan-out (n° max. di ingressi collegabili all'uscita)
- velocità di commutazione



SN7432



SN7400



SN7498

