29.02.2016

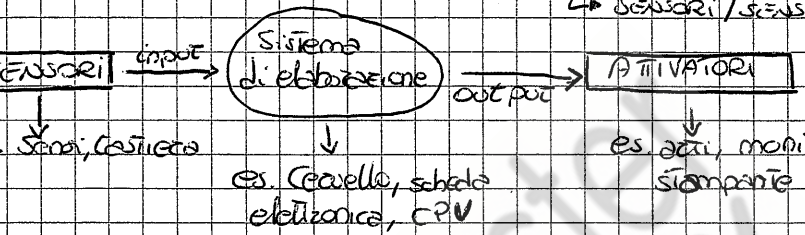
Introduzione

Il problema del calcolatore è il linguaggio di PROGRAMMAZIONE. Alcuni calcolatori sono: il computer, il cellulare, lo smartphones, il tablet ... Il calcolatore segue delle leggi descritte da programmi costituiti da algoritmi che sono il linguaggio del calcolatore. I calcolatori girano mediante un PROCESSORE.

ATTENZIONE: Bisogna verificare la legittimità delle operazioni (algoritmi) che il calcolatore esegue "leggendo" il programma.

Studieremo strumenti di programmazione che sono lo scheletro dei programmi (oggetto di esame!!!). Linguaggio: Java Script. Molti siti web girano su Java che è la base del nostro corso.

Struttura di un sistema di elaborazione: software - LINGUAGGIO DI PROGRAMMAZIONE (e altro)
hardware - PROCESSORE (e altro)
sistema antropomorfo (esistono molti parallelismi)
↳ sensori / sensi



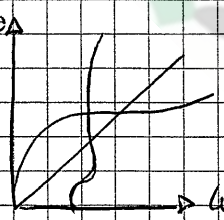
Le Reazioni / RISPOSTE del sistema di elaborazione sono caratterizzate da due modalità di reazione:

(I) SENZA MEMORIA - COMBINATORIA:
- esegue sempre la stessa reazione (es. luce che si accende/spegne)

(II) CON MEMORIA
- complessi e difficili da controllare; utilizzano come base un sistema combinatorio (es. luce del condominio)

I sensori

Percepiscono grandezze continue (es 1. occhio, livelli di luminosità)
(es 2. dita sullo schermo)

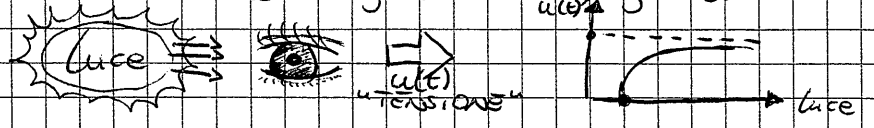


INPUT ricevuti sotto forma di GRANDEZZE CONTINUE, così come gli OUTPUT = RISPOSTE

MA queste grandezze sono difficili da CODIFICARE poiché variano di rumore/incertezze e poiché si esprimono in segnali analogici
Serve un'informazione qualitativa, "discreta" che possa avere 4, 2, ... valori (es. schermo pigiato o no)

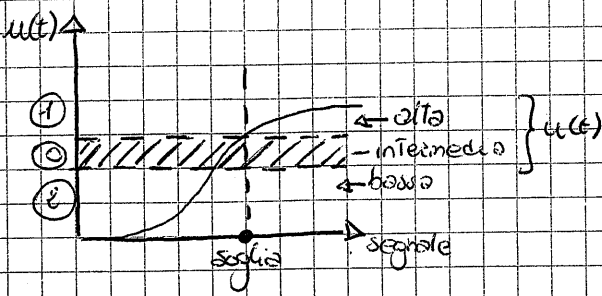
SEGNALI ANALOGICI → SEGNALI DIGITALI

I sistemi elettronici usano segnali DIGITALI, segnali discreti.



? Il problema non è quanta luce, ma c'è abbastanza luce o meno?
↳ OCCORRE un nuovo SENSORE

master COPY COPISTERIA



NB: In 0 c'è indecisione ma la curva è rapida pertanto non significativa

OTTENIAMO Riconoscimento di due segnali (per 1 e 2)

Qualunque tensione → SISTEMA ← TOLLERANZA (anche in calcolo)

↳ poiché usiamo logica = 2 segnali "DIGITALI" → BINARI → (es 0,1 / vero, falso)

Lo stesso vale per: SIST. ELABORAZIONE → ATTUATORI

Classificazione circuiti / S. elaborazione

S | uscita 0 / entrata 0 } INUTILE | S | → uscita 0 / entrata 1 "GENERATORE DI <COSTANTI"

1 → 1 | ∅ → ∅

es luce 1 → ⊗

- **1 uscita 1 entrata**

i → [ ] → o

i può assumere due valori
o può assumere quattro possibili uscite

Ci serviamo della "TABELLA DI VERITÀ"
↳ rappresentazione del comportamento di un sistema combinatorio

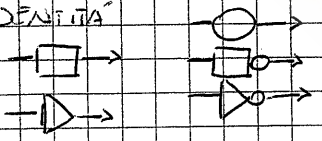
VALORI INPUT i	VALORI OUTPUT o			
0	0	0	1	1
1	0	1	0	1

ogni riga è uno stato di ingresso!

↳ generatore costante 1 →

∅ → generatore di costante ↓ ↳ INVERTITORE

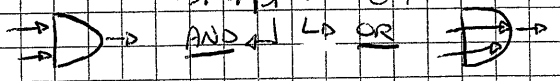
IDENTITÀ



- **2 ingressi 1 uscita**

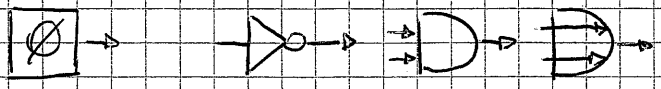
A → [ ] → X
B →

A	B	X	
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1



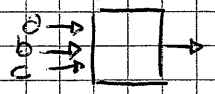
AND ↳ OR

Ricapitolando abbiamo ottenuto:



NB: il numero degli stati di ingresso è 2^n dove n è il # di ingressi
- 1 ing → 2 stati ing
- 2 ing → 4 stati ing

- 3 ingressi 1 uscita

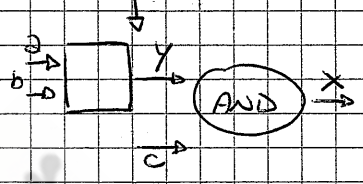


"poniamo la condizione che x sia vera se a, b, c sono veri"

Sono incapace di studiarlo a 3 ingressi, uso *trucchetto*

A	B	C	X	Y
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	1
1	1	1	1	1

PRETESSA



NB: una delle configurazioni stato di ingresso deve dare 1. "PORTA AND"

I) Per trovare Y considero gli stati in cui C è 1.
II) so che Y e C mi danno X. Ricavo gli indecisi — Ottengo

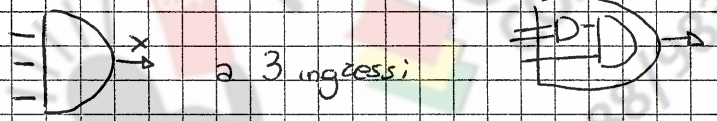
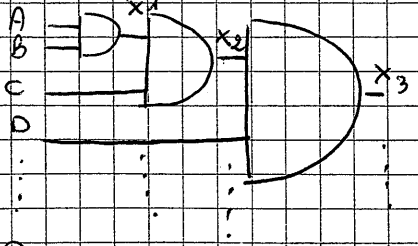


Tabella di verità completa

i_0	i_1	i_2	...	i_{n+1}	X
1	1	1	...	1	1
0	1	0	...	0	1
1	1	0	...	0	1
⋮	⋮	⋮		⋮	⋮
OTHERS					∅

And e Or (a più ingressi)



- se X_1 è vera, A, B sono veri
- se X_2 è vera, A, B, C sono veri
- se X_3 è vera, A, B, C, D sono veri

Per Or sarà analogo

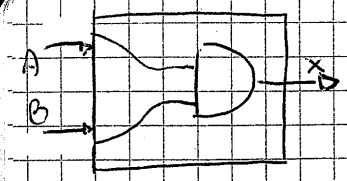
(Ex) Per A e B voglio costruire una porta "and" così

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

ma una porta AND così fatta dovrebbe rendere una X del Tipo

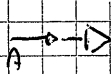
X
0
0
0
1

che non concorde con la X del problema

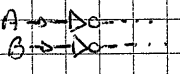


A	X
1	0
1	1
0	0
0	0

se inverto solo A

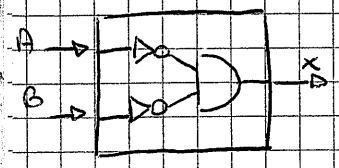


se inverto A e B

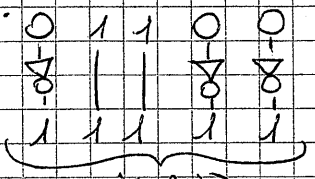


A	B	X
1	1	1
1	0	0
0	1	0
0	0	0

Risultato circuito



NOTA:

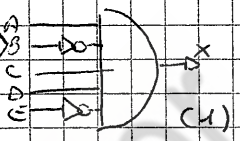


(Ex)

A	B	C	D	E	X
1	0	1	1	0	1
others					Ø



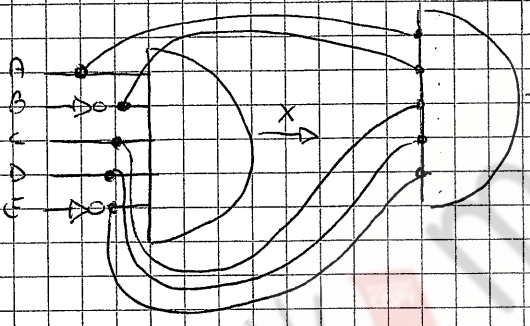
(0)



(1)

A	B	C	D	E	X	Y
1	0	1	1	0	1	0
1	0	1	0	0	0	1
others					0	0

(1)

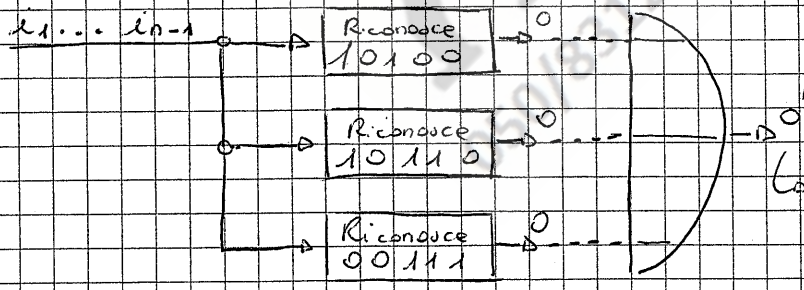


☐ OR – valgono gli stessi ragionamenti

Tabella di verità:

i_0	i_1	...	i_{n-1}	X
0	0	...	0	Ø
others				1

• Ipotizziamo PORTA OR COMBINATA



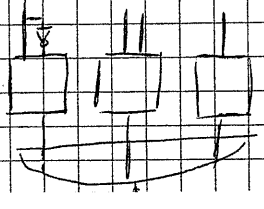
→ se tutti i circuiti collegati sono uguali a zero

↳ MA diventa 1 se almeno uno è diverso da zero

NOTA: Ricordo circuiti astratti (and e inverter) e sintesi mediante somma dei prodotti (Questione)

→ Dati n ingressi e 1 di uscita:
- quanti inverter → N
- quante and (con n ingressi) → 2^n
- quante or → 1 (con 2^n ingressi)

→ Con così tante risorse conviene semplificare



semplificazione

A	0	0	1	1	mi rimane
B	0	0	1	1	1 sola eq.
others				Ø	

STRUTTURARE: significa ridurre i tempi di reazione

Infatti: segn. elettrici → cercare capacità nel sistema → REATTANZA

Vediamo in base alla composizione del circuito

NB: Esempio pratico sul quaderno vecchio

4.03.2016

Riassumendo ...

Circuiti COMBINATORI
- generatori di costante
- 1 ingresso 1 uscita
- 2 ingressi - AND e OR
  - cascata lineare
  - albero, torneo

Definire operazioni che possono semplificare schemi del circuito
↳ addizione, sottrazione, moltiplicazione
↳ Aritmetica → numeri elementari

Logica Booleana

2 valori $\begin{cases}0\\1\end{cases}$ → operatori dovrebbe essere $\begin{cases}\sqcap\to\\ -\!\triangleright\!\circ\to\\ =\!\!\supset\to\end{cases}$
↳ COSTANTI

(Es) $a = \bar{a}^2 + 1$ | a "variabile booleana" = 0 v 1
→ Per risolverla uso degli operatori

Operatori:

- $\bar{a}$, $!a$, $/a$: NEGAZIONE $-\!\triangleright\!\circ-$

i	$\bar{i}$
0	1
1	0

→ $\overline{(\bar{a})} = !!a = //a =$

a	b	c
0	1	0
1	d	1

$= a$

- AND = prodotto logico = ab $a \cdot b$ $a \& b$ $a * b$

A	B	Y	
0	0	0	somme \ prodotto
0	1	0	prodotto
1	0	0	prodotto
1	1	1	prodotto

→ come si compone AND con delle variabili/costanti

$0 \cdot 0 \to \emptyset$	$0 \cdot a \to \emptyset$ ← ENUMERAZIONE
$1 \cdot 1 \to 1$	$1 \cdot a \to a$ ↙ NE
$0 \cdot 1$	
$1 \cdot 0 \to \emptyset$	$a \cdot a \to a$
"COSTANTI"	$a \cdot \bar{a} \to \emptyset$ a e $/a$ variabile

→ AND è commutativo, associativo

- OR "somma logica"

A	B	Y	
0	0	0	← S/P
0	1	1	← S
1	0	1	← S
1	1	1	← P

$a + b$
~ proprietà

$0+0 \to 0$	$0 + a \to a$
$0+1 \to 1$	$1 + a \to 1$
$1+0 \to 1$	
$1+1 \to 1$	$a + a \to a$
	$a + \bar{a} \to 1$

master COPY COPISTERIA