

## introduzione ai circuiti

**Definizione:** I circuiti logici sono realizzati come circuiti integrati realizzati su chip di silicio. Costituito da porta e fili su chip di silicio, inseriti in un package e collegati all'interno con un certo insieme di pin.

Si distinguono per grado di integrazione.

- ▶ SSI : 1-10 porte
- ▶ MSI : 10-100 porte
- ▶ LSI : 100 - 100.000 porte
- ▶ VLSI : > 100.000 porte (CPU completa o più)

In un circuito digitale i valori binari sono ottenuti tramite discretizzazione dei segnali.

**Segnale alto / ingresso:** 1 (vero, voltaggio  $> 1$ )

**Segnale basso / uscita:** 0 (falso, voltaggio  $< 1$ )

**Definizione:** circuito combinatorio → circuito dove lo stato di uscita dipende dalla funzione logica applicata allo stato istantaneo delle sue entrate.

circuito sequenziale → circuito dove lo stato di uscita non dipende solo dalla funzione logica applicata agli ingressi ma anche sulla base di valori pregressi collocati in memoria.

## porte logiche

**Definizione:** porte logiche → componenti elettronici che permettono di svolgere le operazioni logiche primitive oltre che quelle derivate.

Esse realizzano le operazioni principali dell'algebra booleana.

$n \text{ input } (0,1) \rightarrow \text{operazione booleana} \rightarrow 1 \text{ output } (0,1)$

Si divide in → porte logiche fondamentali: AND, OR, NOT

porte logiche derivate: NAND, NOR, XOR

## porte logiche fondamentali

**AND** → prodotto logico



A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

n input → 1 output

**OR** → somma logica



A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

n input → 1 output

**NOT** → negazione logica

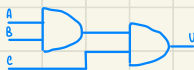


A	$\bar{A}$
0	1
1	0

1 input → 1 output

**OSS:** se possono avere più input, le porte svolgono l'operazione logica associata su N bit  
Nella realizzazione dei circuiti se si hanno a disposizione solo porte a 2 ingressi, vengono collegati a cascata tra loro porte a 2 ingressi.

**ES:** AND a 3 ingressi con 2 AND a 2 ingressi



## porte logiche derivate

Combinazione di porte logiche fondamentali

Scopo: semplificazione dei circuiti

**NAND**  $\rightarrow$  NOT(AND)

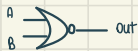


A	B	$A \cdot B$	$A \text{ NAND } B$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

$$A \text{ NAND } B = \text{NOT}(A \text{ AND } B) = (\text{NOT } A) \text{ OR } (\text{NOT } B)$$

$$A \text{ NAND } 1 = \text{NOT}(A \text{ AND } 1) = (\text{NOT } A)$$

**NOR**  $\rightarrow$  NOT(OR)



A	B	$A + B$	$A \text{ NOR } B$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

**XOR**  $\rightarrow$  disgiunzione esclusiva (vero quando A e B sono diversi)



A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

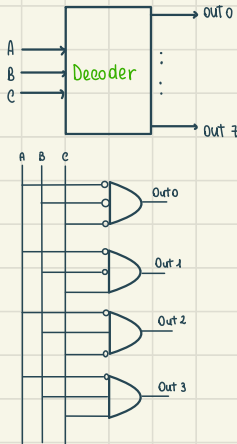
**OSS**: NOR e NAND svolgono la funzione di **inverter**, sono definite **universali**.

## circuiti notevoli

### Decoder

$n$  ingressi  $\rightarrow 2^n$  uscite (un solo valore attivo)

Scopo imposta a stato alto l'uscita corrispondente alla conversione in base 10 della codifica binaria a  $n$  bit

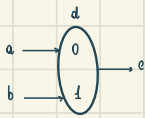


A	B	C	0	1	2	3	4	5	6	7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

### Multiplexor

**Selettore**  $\rightarrow 2^m$  entrate principali  $\rightarrow n$  entrate di controllo  $\rightarrow 1$  output

determina quale input diventa output (pensala come uno switch)



d	c
0	a
1	b

Input  $\rightarrow n$  segnali ( $2^m$ )

Selettori  $\rightarrow m = \log_2 n$  (bit)

**ES:** immaginalo come uno switch che data una serie di segnali (come i casi dello switch), il selettore sceglie tra le opzioni l'output che serve.

### logiche a due livelli

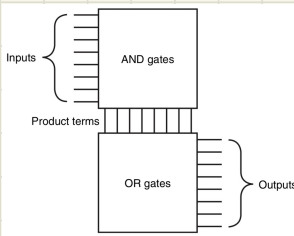
Attraverso le porte logiche AND, OR e NOT è possibile implementare funzioni logiche più complesse

$\rightarrow$  Somma di prodotti (PLA  $\rightarrow$  Programmable Logic Array)

$\rightarrow$  Prodotto di somme

### PLA

- Insieme di input che vengono complementati tramite inverter per gestire più uscite
- Una logica a due stage: array di AND



array di OR

ES: 3 input

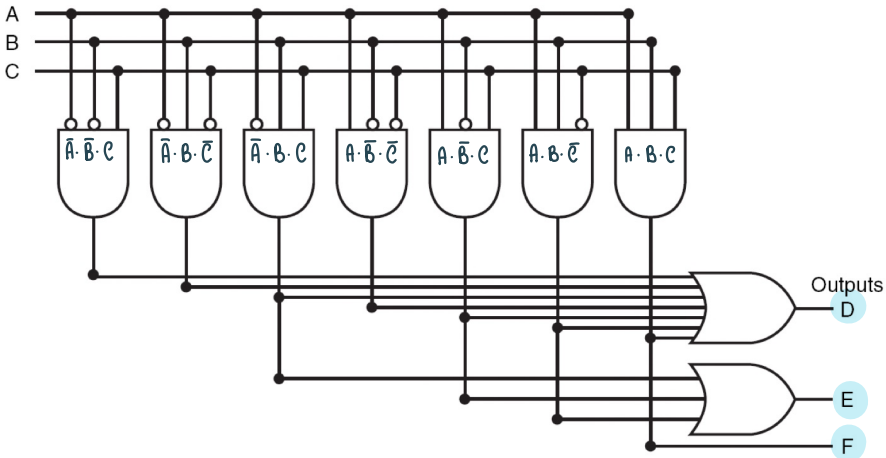
Inputs			Outputs		
A	B	C	D	E	F
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	1	0	0
0	1	1	1	1	0
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	1	0
1	1	1	1	0	1

$$D = (\bar{A} \cdot \bar{B} \cdot C) + (\bar{A} \cdot B \cdot \bar{C}) + (\bar{A} \cdot B \cdot C) + (A \cdot \bar{B} \cdot \bar{C}) + (A \cdot \bar{B} \cdot C) + (A \cdot B \cdot \bar{C})$$

$$E = (\bar{A} \cdot B \cdot C) + (A \cdot \bar{B} \cdot C) + (A \cdot B \cdot \bar{C})$$

$$F = (A \cdot B \cdot C)$$

Inputs



$\bar{A} \cdot \bar{B} \cdot \bar{C}$  non c'è  
perché non ha output

**Definizione:** un bus è una collezione di input trattati come un unico segnale (molte operazioni sono gestite su 32 bit. In questo modo un multiplexor con un bus da 32 bit si comporta come un array da 32 multiplexor ad 1 bit ciascuno.