

Porte Logiche Logic Gates

Corrado Santoro

Dipartimento di Matematica e Informatica
santoro@dmi.unict.it



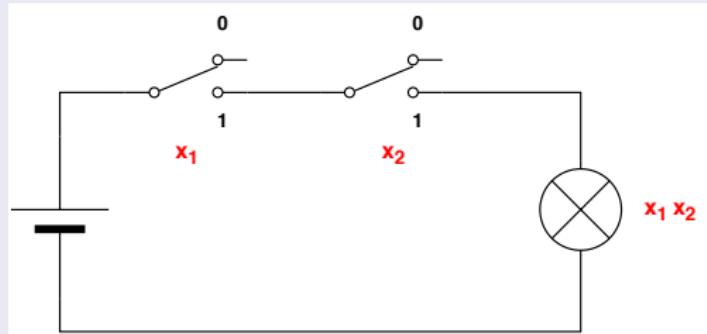
Corso di Architettura degli Elaboratori

Logic Gates

- Le **Logic Gates** sono dei circuiti elettronici che realizzano (implementano) le operazioni dell'**algebra booleana**
- Esse operano mappando i concetti 0 e 1 sul principio elettrico **OFF/ON**
- Sono costituite da un rete di componenti elettronici denominati **transistor**, realizzati tramite tecnologia **MOS = Metal-Oxide-Semiconductor**
- Ogni transistor si comporta come **interruttore controllato**; una logic gate è dunque una rete di **interruttori controllati**
- Tramite tali reti è possibile implementare molto facilmente qualunque funzione dell'**algebra booleana**

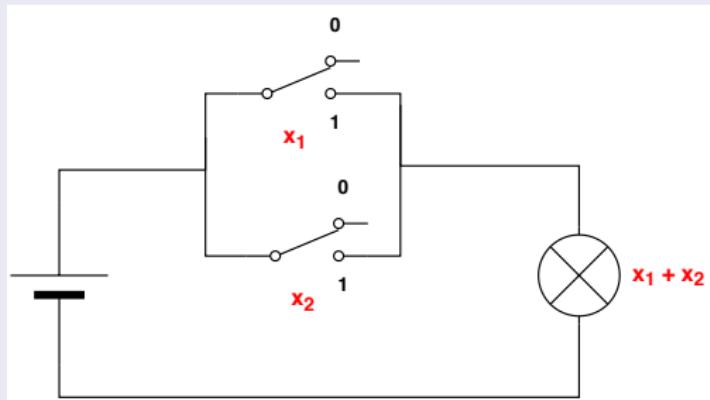
Prodotto Logico—AND

- Due interruttori collegati in serie possono rappresentare il prodotto logico
- La lampadina si accende (“1”) quanto **entrambi** gli interruttori sono in posizione “1”



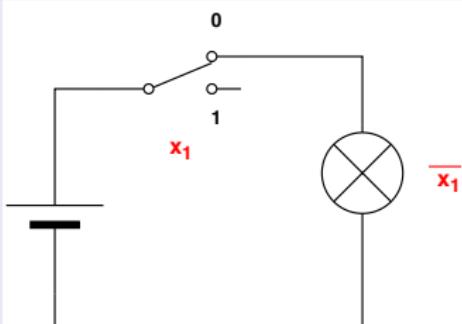
Somma Logica—OR

- Due interruttori collegati in parallelo possono rappresentare la somma logica
- La lampadina si accende (“1”) quanto **almeno uno dei due** interruttori è in posizione “1”



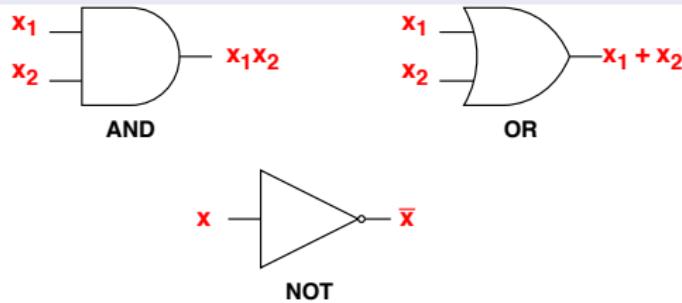
Negazione Logica—NOT

- La lampadina si accende (“1”) quanto l'interruttore è in posizione “0”



Porte Logiche

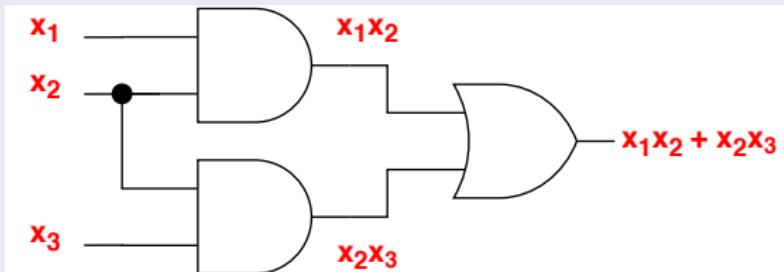
- I circuiti elettronici basati su algebra booleana utilizzano dei simboli specifici per rappresentare le “porte logiche” che implementano le operazioni base:
 - Prodotto logico: **AND gate**
 - Somma logico: **OR gate**
 - Negazione: **NOT gate**
- Una qualunque funzione logica viene implementata elettronicamente interconnettando opportunamente varie logic gates



Esempio di funzione logica

- Una qualunque funzione logica viene implementata elettronicamente interconnettando opportunamente varie logic gates

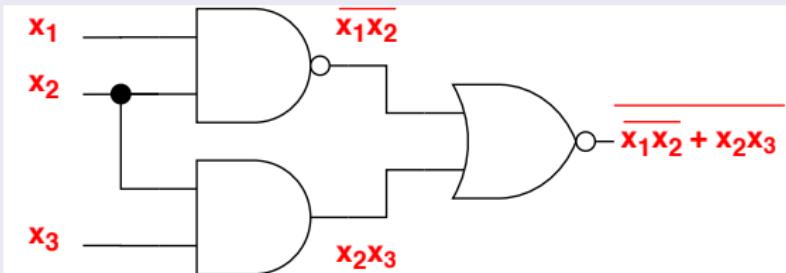
$$y = x_1 x_2 + x_2 x_3$$



Esempio di funzione logica

- Qualora il NOT sia associato all'ingresso o all'uscita di una porta AND o OR, il simbolo si abbrevia con un semplice pallino

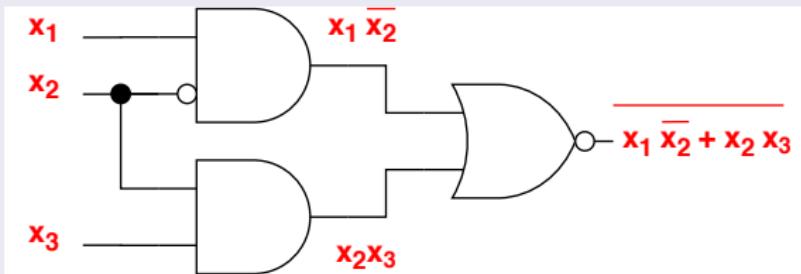
$$y = \overline{\overline{x_1x_2} + x_2x_3}$$



Esempio di funzione logica

- Qualora il NOT sia associato all'ingresso o all'uscita di una porta AND o OR, il simbolo si abbrevia con un semplice pallino

$$y = \overline{x_1 \bar{x}_2 + x_2 x_3}$$



Teoremi di De Morgan

$$\overline{X_1} \ \overline{X_2} = \overline{X_1 + X_2}$$



$$\overline{X_1} + \overline{X_2} = \overline{X_1 \cdot X_2}$$



Implementazione della **Somma Aritmetica** tramite porte logiche

Somma Binaria Aritmetica

Regole della Somma Binaria Aritmetica

x_1	x_2	Somma Aritmetica S	Riporto (Carry) C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Somma Binaria

x_1	x_2	Somma Aritmetica S
0	0	0
0	1	1
1	0	1
1	1	0

Riporto

x_1	x_2	Riporto (Carry) C
0	0	0
0	1	0
1	0	0
1	1	1



Somma Binaria Aritmetica

Somma Binaria

x_1	x_2	Somma Aritmetica S
0	0	0
0	1	1
1	0	1
1	1	0

$$S = \overline{x_1}x_2 + x_1\overline{x_2}$$

Riporto

x_1	x_2	Riporto (Carry) C
0	0	0
0	1	0
1	0	0
1	1	1

$$C = x_1x_2$$

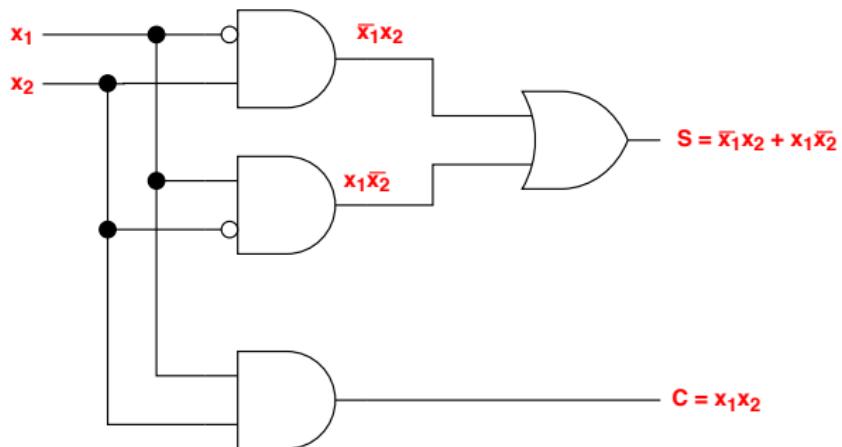
Somma Binaria Aritmetica

Somma Binaria

$$S = \bar{x}_1 x_2 + x_1 \bar{x}_2$$

Riporto

$$C = x_1 x_2$$

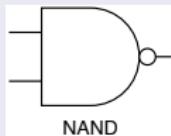


Half-Adder

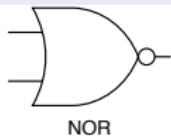
Porte Logiche Derivate

Porte Logiche Derivate

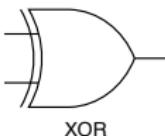
- **NAND gate** ⇒ $y = \overline{x_1 x_2}$
- **NOR gate** ⇒ $y = \overline{x_1 + x_2}$
- **XOR gate** (OR-esclusivo) ⇒ $y = \overline{x_1}x_2 + x_1\overline{x_2}$
- **XNOR gate** (NOR-esclusivo) ⇒ $y = \overline{\overline{x_1}x_2 + x_1\overline{x_2}}$



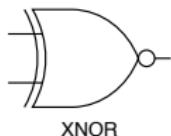
NAND



NOR



XOR



XNOR

Porte Logiche Derivate

XOR

x_1	x_2	$\overline{x_1}x_2 + x_1\overline{x_2}$
0	0	0
0	1	1
1	0	1
1	1	0

XNOR

x_1	x_2	$\overline{\overline{x_1}x_2 + x_1\overline{x_2}}$
0	0	1
0	1	0
1	0	0
1	1	1

Porte Logiche Logic Gates

Corrado Santoro

Dipartimento di Matematica e Informatica
santoro@dmi.unict.it



Corso di Architettura degli Elaboratori