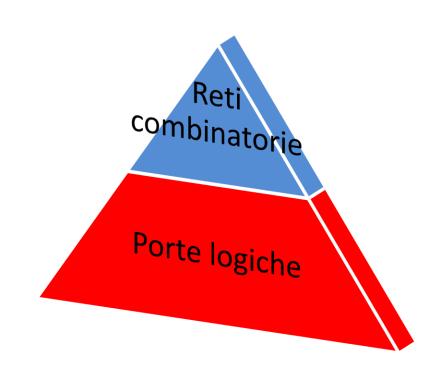


### ARGOMENTI DELLA LEZIONE

- ☐ Porte logiche
- Reti combinatorie
  - ☐ Codificatore e Decodificatore
  - ☐ Addizionatore e Sottrattore
  - ☐ Comparatore aritmetico e logico





#### Generalità

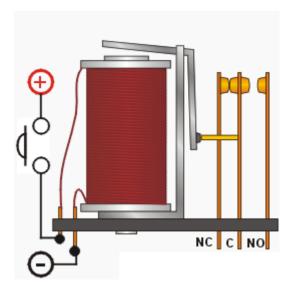
- Una **porta logica** (*gate*) è un elemento di calcolo, realizzato mediante un componente elettromeccanico (relé, anni 1920-1940) o elettrico (*transistor*, dal 1950) avente un determinato numero di **linee di ingresso** (*fan-in*) ed una **linea di uscita** (*fan-out*) che, eventualmente, può essere collegata all'entrata di una o più porte (eccetto quella da cui esce)
- ☐ I segnali applicati alle linee di ingresso e di uscita sono segnali elettrici e si possono associare a essi due valori convenzionali:
  - ☐ 1 (presenza di segnale o segnale alto: [2V;5V])
  - □ 0 (assenza di segnale o segnale basso: [0V;1V])

# A Second

#### **PORTE LOGICHE**

#### Tecnologia passata: i relè

- Il **relè** è un componente elettromeccanico costituito da una bobina di filo conduttore elettrico, generalmente di rame, avvolto intorno ad un nucleo di materiale ferromagnetico
- Al passaggio di corrente elettrica nella bobina, l'elettromagnete attrae l'ancora alla quale è vincolato il contatto mobile che quindi cambia posizione
- ☐ Un relè è utilizzato per controllare un circuito elettrico, interrompendo o stabilendo il flusso di corrente in risposta a un segnale di controllo. In sostanza, funziona come un interruttore che può essere attivato o disattivato da un'altra fonte di energia o segnale elettrico



## A Second

#### PORTE LOGICHE

#### Tecnologie usate

- Le tecnologie microelettroniche oggi più usate per la realizzazione di porte logiche sono:
  - BJT (Bipolar Junction Transistor): transistor bipolari
    - ☐ TTL (transistor-transistor logic)
    - ☐ ECL (emitter coupled logic)
  - MOS (field-effect transistor)
    - □ pMOS
    - ☐ nMOS
    - CMOS

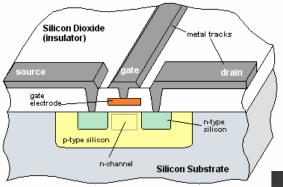
TECNOLOGIA	PRO	CONTRO
ВЈТ	Veloci	Consumano energia Dimensione superiori rispetto alla tecnologia MOS
MOS	Richiedono meno energia Più piccole	Lente

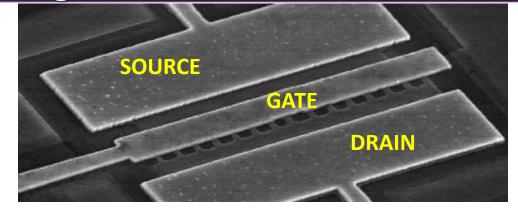
# A SE

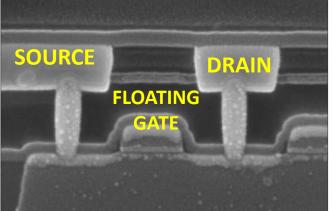
### **PORTE LOGICHE**

Tecnologie usate: nMOS

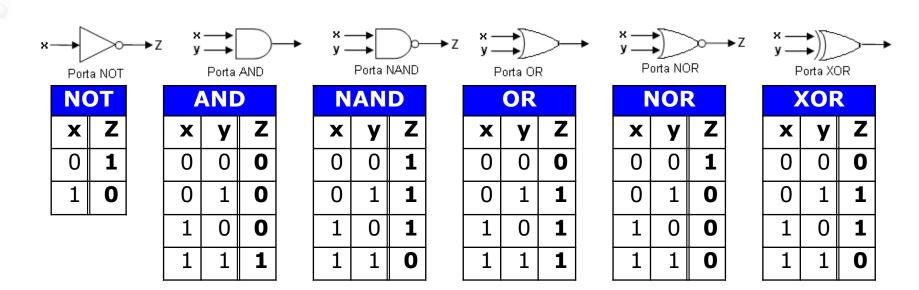
#### NMOS Transistor (n-channel MOSFET)





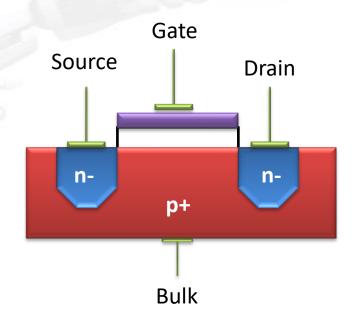


- ☐ Ciascuna porta logica risolve una funzione (o tabella della verità)
- ☐ Le **principali porte logiche** utilizzate sono: NOT, OR, NOR, AND, NAND e XOR



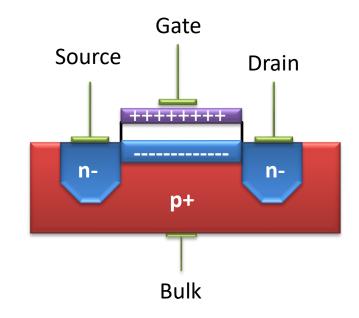


#### nMOS



Nessun traffico di corrente

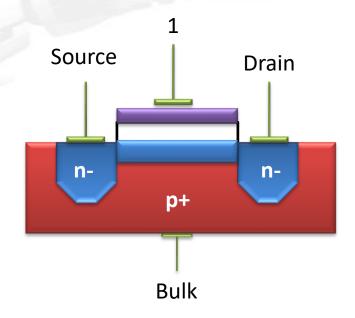
Applicazione di alta tensione al gate



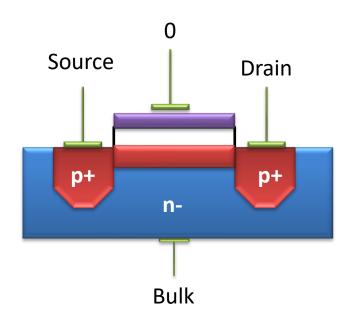
Traffico di corrente da source a drain



CMOS: nMOS - pMOS

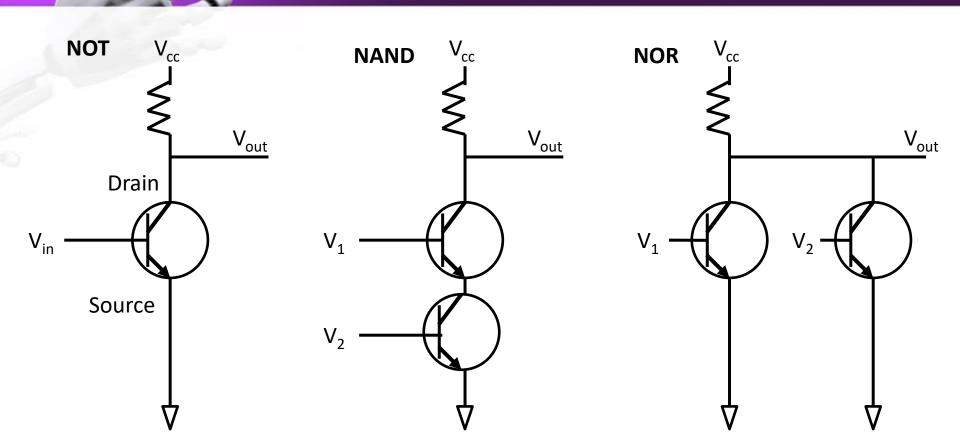


Nessun traffico di corrente

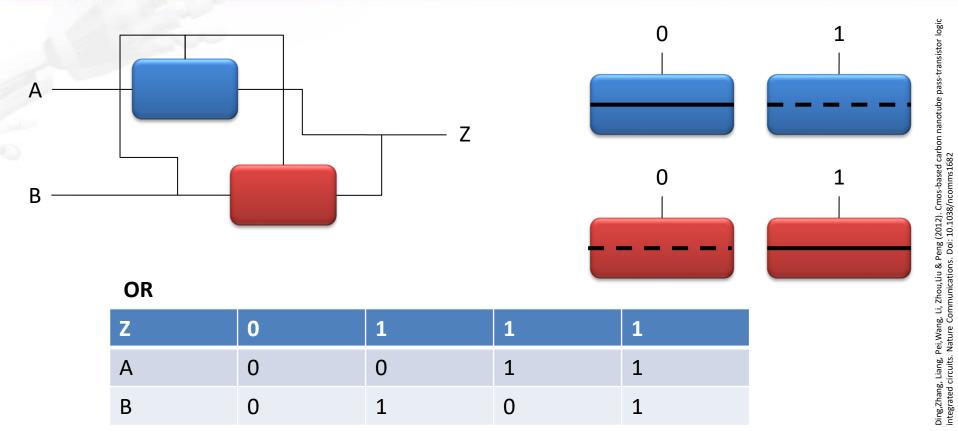


Traffico di corrente da source a drain

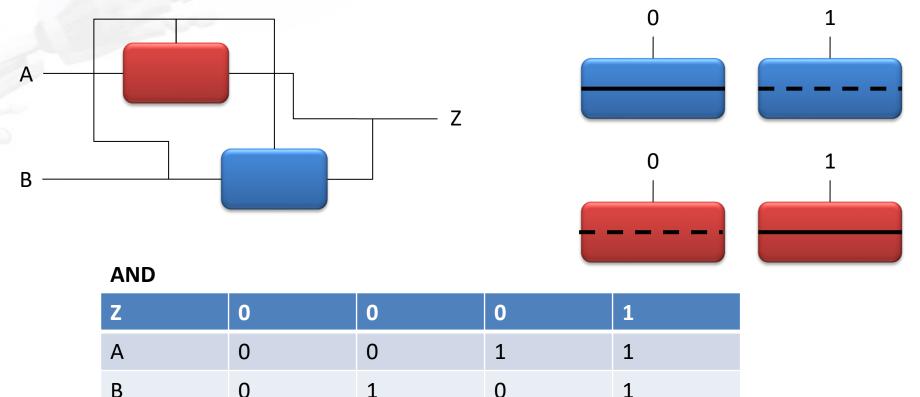
Porte logiche: rappresentazione canonica



### Porte logiche: funzionamento CMOS



### Porte logiche: funzionamento CMOS



Ding,Zhang, Liang, Pei,Wang, Li, Zhou,Liu & Peng (2012). Cmos-based carbon nanotube pass-transistor logic integrated circuits. Nature Communications. Doi: 10.1038/ncomms1682



#### Considerazioni

"L'ENIAC è dotato di 18000 valvole e pesa 30 tonnellate, i calcolatori del futuro potranno avere 1000 valvole e forse peseranno solo una tonnellata e mezza"

Popular Mechanics, Marzo, 1949

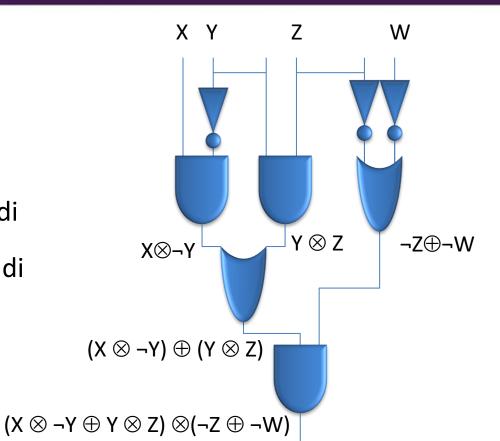


	Nomenclatura	Num. Transitori
SSI	Small Scale Integration	meno di 10transistor
MSI	Medium Scale Integration	da 10 a 100transistor
LSI	Large Scale Integration	da 100 a 10000transistor
VLSI	Very Large Scale Integration	da 10000 a 100.000transistor
ULSI	Ultra Large Scale Integration	fino a 10.000.000 di transistor



#### Generalità

- ☐ Una **rete** è una interconnessione di componenti attivi,le **porte**, collegati tra loro mediante componenti passivi, **le linee**
- ☐ Una rete combinatoria è un circuito elettronico in grado di elaborare, in modo automatico, funzioni binarie di una o più variabili binarie



#### **Definizione**

☐ Formalmente una **rete combinatoria** è definita come un

dispositivo con *n* linee di ingresso

ed *m* linee di uscite per cui i segnali

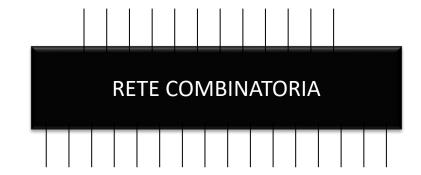
di uscita dipendono unicamente dai

segnali di ingresso

$$Z = F(x)$$
 con  $F : \{0,1\}^n \to \{0,1\}^m$ 

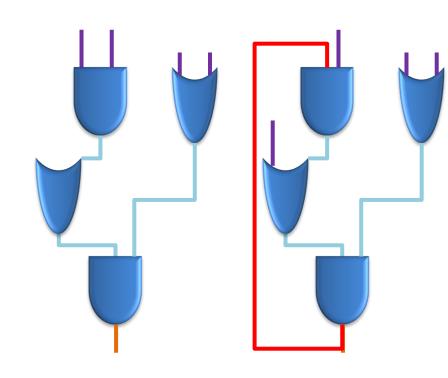
*Osservazione*. In una rete combinatoria non sono presenti cicli né cappi

Nelle reti combinatorie non può avvenire che gli stessi ingressi forniti in istanti diversi diano luogo ad uscite diverse



## RETE COMBINATORIA Schema circuitale

- Uno **schema circuitale** è un collegamento di porte (rappresentate in maniera grafica) tramite linee.
- Ci sono tre tipi di linee:
  - **1. linee di ingresso**, ognuna etichettata con una delle n variabili booleane
  - 2. linee di uscita, ognuna etichettata con una delle m variabili di uscita
  - **3. linee interne**, ciascuna delle quali collega l'uscita di una porta con l'ingresso di un'altra porta.
- ☐ Vincoli:
  - Ogni ingresso di ogni porta deve essere collegato ad una linea di ingresso oppure ad una linea interna.
  - L'uscita di ogni porta deve essere collegata ad una linea di uscita oppure ad una linea interna
  - ☐ Il collegamento di porte tramite linee non deve dare luogo a cicli.



#### Relazione

Una rete combinatoria può essere vista come un dispositivo in grado di soddisfare una tabella, la **tabella della verità**, che per ognuna delle  $2^n$  combinazioni possibili relative agli n valori di ingresso  $(x_1,x_2,...,x_n)$  indica gli m valori di uscita  $(z_1,z_2,...,z_m)$ 

**Osservazione.** Ad ogni rete caratterizzata da una tabella con *n* ingressi ed *m* uscite corrisponde un gruppo di *m* espressioni booleane (e viceversa)



#### **Esempio**

Realizzare un circuito capace di confrontare 2 numeri interi N1 e N2 senza segno a 2 bit, tale circuito fornisce in uscita il valore 1 se tale circuito fornisce in uscita il valore 1 se N1N2, e 0 altrimenti

Z = 1 se N1 >= N2

Z = 0 se N1 < N2

**Esempio** 

N1 (MSB)	N1 (LSB)	N2 (MSB)	N2 (LSB)	Z
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

Derivazione dei maxtermini:

 $\neg N1M \otimes \neg N1L \otimes N2M \otimes \neg N2L \oplus$ 

 $\neg$ N1M $\otimes$  $\neg$ N1L $\otimes$ N2M $\otimes$ N2L $\oplus$ 

 $\neg$ N1M $\otimes$ N1L $\otimes$ N2M $\otimes$  $\neg$ N2L $\oplus$ 

 $\neg$ N1M $\otimes$ N1L $\otimes$ N2M $\otimes$ N2L $\oplus$ 

 $N1M \otimes \neg N1L \otimes \neg N2M \otimes N2L \oplus$ 

 $N1M\otimes \neg N1L\otimes N2M\otimes N2L\oplus$ 

 $N1M \otimes N1L \otimes \neg N2M \otimes N2L \oplus N1M \otimes N1L \otimes N2M \otimes N2L$ 

#### **Esempio**

$$\overline{N1M} \cdot \overline{N1L} \cdot \overline{N2M} \cdot \overline{N2L} + \overline{N1M} \cdot \overline{N1L} \cdot N2M \cdot N2L + \overline{N1M} \cdot N1L \cdot N2M \cdot \overline{N2L} \hspace{0.2cm} \oplus$$

- $\oplus \ \ \overline{N1M} \cdot N1L \cdot N2M \cdot N2L + N1M \cdot \overline{N1L} \cdot \overline{N2M} \cdot N2L + N1M \cdot \overline{N1L} \cdot N2M \cdot N2L$
- $\oplus \ \ N1M \cdot N1L \cdot \overline{N2M} \cdot N2L + N1M \cdot N1L \cdot N2M \cdot N2L$

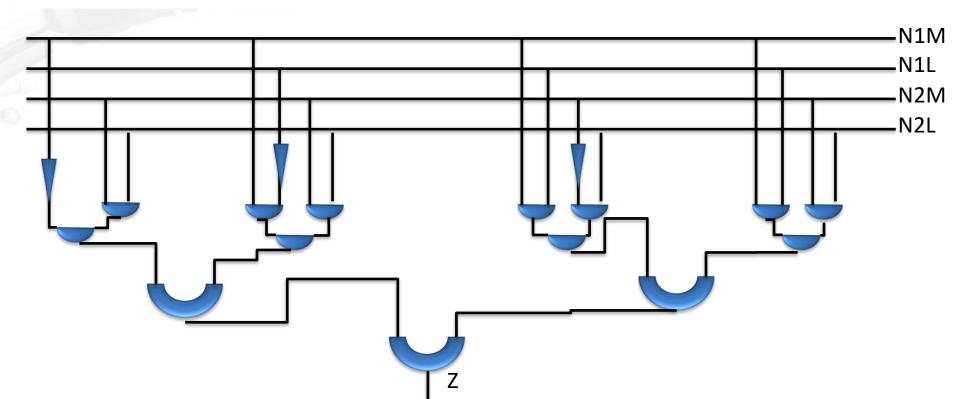
Mediante del tecniche di ottimizzazione si ottiene:

 $\overline{N1L} \cdot N2M \cdot N2L \oplus N1M \cdot \overline{N1L} \cdot N2M \cdot N2L \oplus N1M \cdot N1L \cdot \overline{N2M} \cdot N2L \oplus N1M \cdot N1L \cdot N2M \cdot N2L$ 



#### Esempio

 $\overline{N1L} \cdot N2M \cdot N2L \oplus N1M \cdot \overline{N1L} \cdot N2M \cdot N2L \oplus N1M \cdot N1L \cdot \overline{N2M} \cdot N2L \oplus N1M \cdot N1L \cdot N2M \cdot N2L$ 



#### Caratteristiche progettuali

- Tra i parametri principali nel progetto di una rete combinatoria, è opportuno considerare:
  - l'assorbimento di energia (che stabilisce un limite complessivo al numero di porte utilizzabili)
  - il *ritardo* (che determina la velocità di calcolo)
    - ☐ La velocità di calcolo della rete combinatoria varia in base alla profondità della rete (di solito si considera un tempo costante perché la rete combinatoria ha una profondità finita)
  - ☐ il *costo* di realizzazione
    - ☐ Dipende dal numero di transistor impiegati che cambia a seconda della tecnologia usata, della funzione da soddisfare e il numero di ingressi

es.: la porta NOT è costituita da 1 transistor, NAND o NOR 2 transistor; AND e OR 3 o 4 transistor; altre porte: > 4 transistor

PORTA	TEMPO DI COMMUTAZIONE DEL SEGNALE
NAND, NOR	ΔΤ
AND, OR	2∆T
XOR	<b>4</b> ΔT

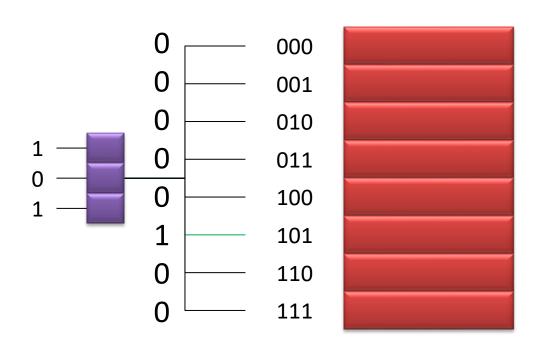
#### **Decodificatore**

- Il decodificatore è una rete combinatoria che trasforma parole associate a codifiche strette in parole associate a codifiche lasche (le linee di uscita sono in numero maggiore rispetto le linee di ingresso)
- Un decodificatore è una rete combinatoria con m linee di ingresso e  $n=2^m$  linee di uscita
- ☐ Logicamente il decodificatore riconosce una stringa (es.: una locazione di memoria o una istruzione)

	DECODIFICATORE									
Line	e ingr	esso				Linee	uscita			
X <sub>2</sub>	X <sub>1</sub>	X <sub>0</sub>	<b>Z</b> <sub>7</sub>	z <sub>6</sub>	<b>Z</b> <sub>5</sub>	Z <sub>4</sub>	Z <sub>3</sub>	Z <sub>2</sub>	Z <sub>1</sub>	z <sub>o</sub>
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

Decodificatore: esempio di impiego

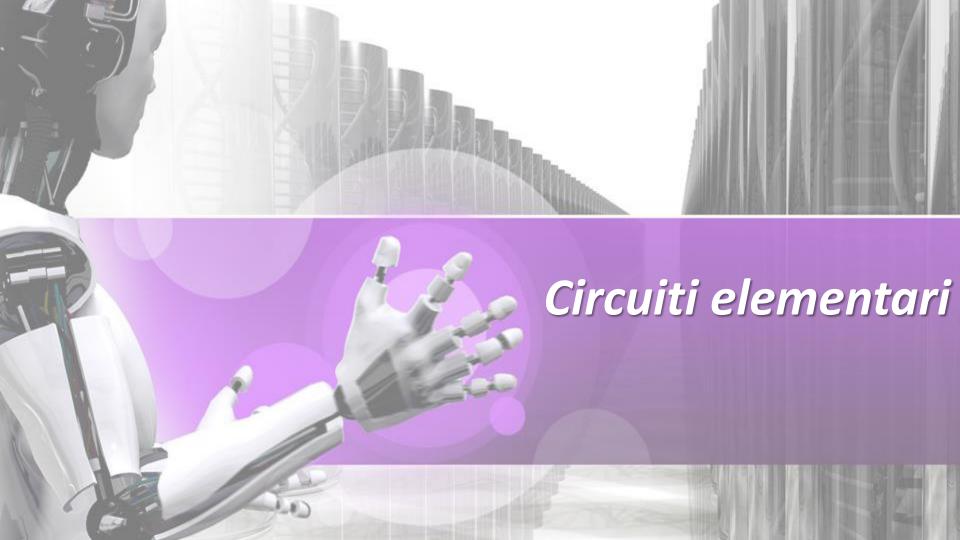
☐ Il decodificatore è usato, ad esempio, per identificare una cella di memoria



#### **Codificatore**

- I codificatori sono una famiglia di reti combinatorie che trasformano parole codificate in una codifica lasca in parole d'uscita rappresentate in codifica stretta
- In generale un codificatore è una rete combinatoria che ha *n* linee di ingresso e *m*= log₂(*n*) linee di uscita; cioè vi è la produzione della codifica binaria dell'indice dell'unica linea di ingresso attiva
- Logicamente il codificatore è un generatore di codici (es.: dei comandi)

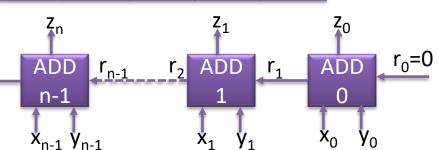
	CODIFICATORE									
	Linee ingresso									cita
X <sub>7</sub>	x <sub>6</sub>	<b>X</b> <sub>5</sub>	X <sub>4</sub>	<b>X</b> <sub>3</sub>	X <sub>2</sub>	X <sub>1</sub>	x <sub>o</sub>	   <b>Z</b> 2	Z <sub>1</sub>	z <sub>o</sub>
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1



#### **Addizionatore**

- L'addizionatore è una rete combinatoria che consente l'operazione di somma tra due operandi (addendi)
- La rete combinatoria associata ad un addizionatore può essere realizzata mediante una tecnica di decomposizione. Questo perché l'addizione di due numeri binari può essere vista come la somma di due bit alla *i*-esima posizione (x<sub>i</sub> e y<sub>i</sub>) ai quali va aggiunto il **riporto** (r<sub>i</sub>) per ottenere un **risultato** (z<sub>i</sub>) ed un eventuale **riporto** (r<sub>i+1</sub>) per le **cifre successive**

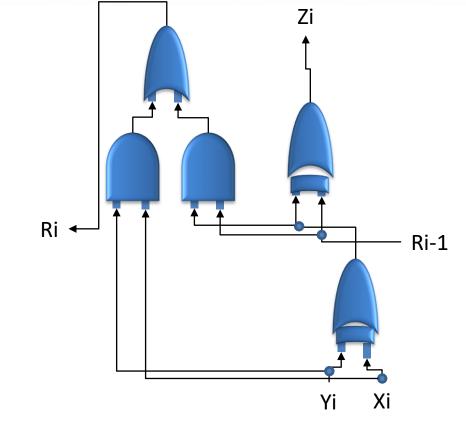
	ADDIZIONATORE							
X <sub>i</sub>	<b>y</b> i	r <sub>i</sub>	z <sub>i</sub>	r <sub>i+1</sub>				
0	0	0	0	0				
0	0	1	1	0				
0	1	0	1	0				
0	1	1	0	1				
1	0	0	1	0				
1	0	1	0	1				
1	1	0	0	1				
1	1	1	1	1				



#### **Addizionatore**

		ADDI	ZIONA	TORE				
	x <sub>i</sub>	<b>y</b> i	r <sub>i</sub>	z <sub>i</sub>	r <sub>i+1</sub>			
	0	0	0	0	0			
	0	0	1	1	0			
	0	1	0	1	0			
	0	1	1	0	1			
	1	0	0	1	0			
	1	0	1	0	1			
	1	1	0	0	1			
	1	1	1	1	1			
	$z_n$ $z_1$ $z_0$							
r	A	DD r <sub>n-</sub>	$_1$ $r_2$	ADD	$r_1$			

n-1

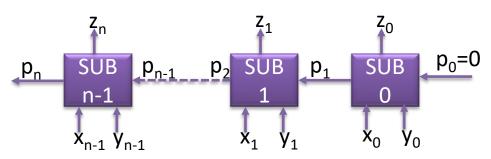


#### **Sottrattore**

- Il sottrattore è una rete combinatoria che permette la sottrazione tra due operandi (minuendo e sottraendo)
- Anche in questo caso è possibile fare riferimento ad una struttura modulare.

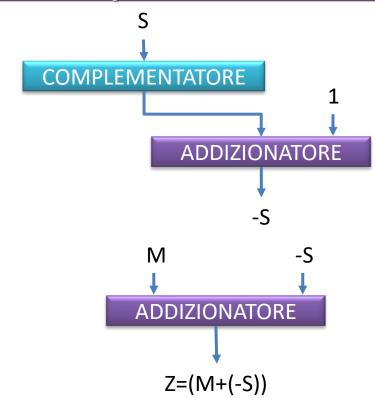
All'i-esimo bit del minuendo  $(x_i)$  va sottratto sia il sottraendo  $(y_i)$  sia il bit di prestito  $(p_i)$  della posizione precedente per poi generare il bit risultante  $(z_i)$  ed il bit del prestito  $(p_{i+1})$  per le cifre successive

SOTTRATTORE							
X <sub>i</sub>	$x_i$ $y_i$ $p_i$ $z_i$ $p_{i+1}$						
0	0	0	0	0			
0	0	1	1	1			
0	1	0	1	1			
0	1	1	0	1			
1	0	0	1	0			
1	0	1	0	0			
1	1	0	0	0			
1	1	1	1	1			



#### Sottrattore: realizzazione pratica

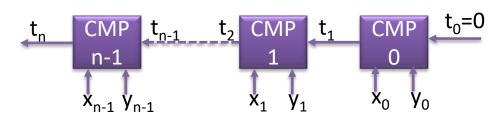
- Nel concreto la realizzazione di un sottrattore può essere effettuato con una circuiteria differente da quella vista
- L'operazione di sottrazione z=m-s si riduce alla espressione equivalente z=(m+(-s))
- ☐ Si utilizza un complementatore ed un addizionatore che prende in input come addendi il numero complementato ed il valore 1
- L'operazione di sottrazione si ottiene aggiungendo il minuendo



#### Comparatore

- Il comparatore è una rete combinatoria che ha una unica linea di uscita che vale 1 se il numero x, di n bit, applicato in ingresso risulta maggiore o uguale (in senso algebrico) al numero y, di n bit, anch'esso preso in ingresso, con cui si effettua il confronto
- Anche per tale componente è possibile fare riferimento ad una struttura modulare

COMPARATORE							
x <sub>i</sub>	<b>y</b> i	t <sub>i</sub>	t <sub>i+1</sub>				
0	0	0	0				
0	0	1	1				
0	1	0	0				
0	1	1	0				
1	0	0	1				
1	0	1	1				
1	1	0	0				
1	1	1	1				



## R

#### RETE COMBINATORIA

#### **Comparatore logico**

- Altresì, invece, è possibile realizzare un **comparatore logico** utilizzando *n* porte XOR, a cui in ingresso sono associati gli i-esimi bit dei valori da comparare, le cui uscite sono collegate ad una porta OR
- ☐ In questo caso, infatti, è necessario stabilire solamente se le stringhe binarie sono uguali o diverse
- ☐ Un comparatore logico determina il risultato finale in tempo costante O(1)

