

4.2 Algebra di commutazione

Algebra di commutazione

1) **Costanti:** 0, 1

2) **Operazioni:**
 somma logica (+) prodotto logico (.) complementazione (')

3) **Postulati:**

$0 + 0 = 0$	$0 \cdot 0 = 0$	$0' = 1$
$1 + 0 = 1$	$1 \cdot 0 = 0$	$1' = 0$
$0 + 1 = 1$	$0 \cdot 1 = 0$	
$1 + 1 = 1$	$1 \cdot 1 = 1$	

4) **Variabili:** simboli sostituibili o con 0 o con 1

Espressioni

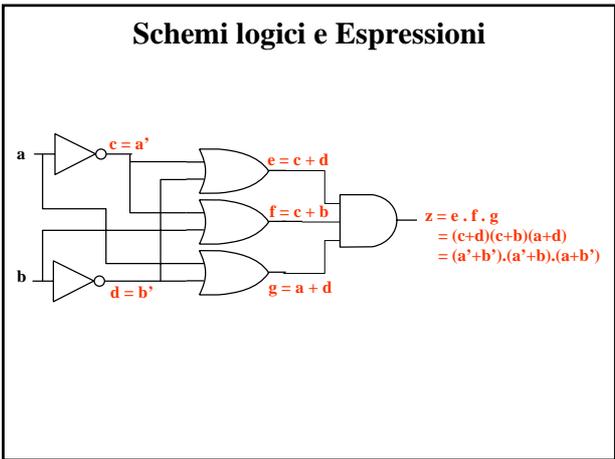
5) **Espressione** - Stringa finita di costanti, variabili, operatori e parentesi, formata in accordo con le seguenti regole:

- 1) 0 e 1 sono espressioni
- 2) una variabile è una espressione
- 3) se A è un'espressione, lo è anche (A')
- 4) se A, B sono espressioni, lo sono anche (A+B), (A.B)

Esempi:

$a \cdot b$	$a + (b \cdot c)$	$(a + b)'$	$a + bc$	$a'b + 0 + ab'$
-------------	-------------------	------------	----------	-----------------

N.B - L'operazione di prodotto è prioritaria rispetto alla somma e non è obbligatorio racchiuderla tra parentesi. La notazione AB indica A.B



Teoremi di equivalenza

Equivalenze notevoli

Proprietà della somma e del prodotto logico:

E1) commutativa	$x + y$	=	$y + x$
	$x \cdot y$	=	$y \cdot x$
E2) associativa	$(x + y) + z$	=	$x + y + z$
	$(x \cdot y) \cdot z$	=	$x \cdot y \cdot z$
E3) distributiva	$(x \cdot y) + (x \cdot z)$	=	$x \cdot (y + z)$
	$(x + y) \cdot (x + z)$	=	$x + (y \cdot z)$
E4) idempotenza	$x + x$	=	x
	$x \cdot x$	=	x
E5) identità	$x + 0$	=	x
	$x \cdot 1$	=	x
E6) limite	$x + 1$	=	1
	$x \cdot 0$	=	0

Equivalenze notevoli

Proprietà della complementazione:

E7) involuzione $(x')' = x$

E8) limitazione $x + x' = 1$

$x \cdot x' = 0$

E9) combinazione $xy + xy' = x$

$(x+y).(x+y') = x$

E10) I^a legge di De Morgan $(x + y)' = x' \cdot y'$

II^a legge di De Morgan $(x \cdot y)' = x' + y'$

E11) consenso $xy + x'z + yz = xy + x'z$

$(x+y).(x'+z).(y+z) = (x+y).(x'+z)$



Espressioni di funzioni incomplete

ENCODER a 3 ingressi

x_2	x_1	x_0	z_1	z_0
0	0	0	0	0
1	0	0	1	1
0	1	0	1	0
0	0	1	0	1

x_2	x_1	x_0	z_1	z_0
0	0	0	0	0
1	0	0	1	1
0	1	0	1	0
0	0	1	0	1
1	1	0	-	-
1	0	1	-	-
0	1	1	-	-
1	1	1	-	-

N.B. le altre configurazioni sono per ipotesi impossibili

Espressioni di funzioni incomplete

Espressioni equivalenti di funzioni incomplete - Espressioni che forniscono eguale valutazione limitatamente al dominio di una funzione incompleta sono dette **equivalenti**.

Espressioni per l'ENCODER:

$z_1 = x_2 x_1' x_0' + x_2' x_1 x_0'$

$z_0 = x_2 x_1' x_0 + x_2' x_1 x_0$

$u_1 = x_2 + x_1$

$u_0 = x_2 + x_0$

x_2	x_1	x_0	z_1	u_1	z_0	u_0
0	0	0	0	0	0	0
1	0	0	1	1	1	1
0	1	0	1	1	0	0
0	0	1	0	0	1	1
1	1	0	0	1	0	1
1	0	1	0	1	0	1
0	1	1	0	1	0	1
1	1	1	0	1	0	1

Espressioni
canoniche

Espressioni canoniche

T6) Espressione canonica SP (Somma di Prodotti)

I^a forma canonica - Ogni funzione di n variabili è descritta da una somma di tanti prodotti logici quante sono le configurazioni per cui vale 1. In ciascuna somma, **maxtermine**, appare ogni variabile, in forma vera se nella configurazione corrispondente vale 1, in forma complementata se vale 0.

T7) Espressione canonica SC (Somma di Prodotti)

II^a forma canonica - Ogni funzione di n variabili è descritta da un prodotto di tante somme logiche quante sono le configurazioni per cui vale 0. In ciascuna somma, **maxtermine**, appare ogni variabile, in forma vera se nella configurazione corrispondente vale 0, in forma complementata se vale 1.

Completezza
dell'algebra
di commutazione

Espressioni canoniche della funzione "a implica b"

a	b	a→b
0	0	1
0	1	1
1	0	0
1	1	1

II^a forma canonica:
 $F(a,b) = a' + b$

I^a forma canonica:
 $F(a,b) = a' \cdot b' + a' \cdot b + a \cdot b$

Verifica della equivalenza per manipolazione algebrica:

$F(a,b) = a' \cdot b' + a' \cdot b + a \cdot b$

$= a' \cdot (b' + b) + a \cdot b$

E3

$= a' \cdot 1 + a \cdot b$

E8

$= a' + a \cdot b$

E5

$= a' + a \cdot b + a' \cdot b$

una parte è inclusa nel tutto

$= a' + b$

E3, E8, E5

Sintesi canonica del EX-OR

x_0	x_1	$x_0 \oplus x_1$
0	0	0
0	1	1
1	0	1
1	1	0

1 se e solo se $x_0=0$ e $x_1=1$
oppure se $x_0=1$ e $x_1=0$
0 negli altri due casi

1 se e solo se $x_0=1$ e $x_1=0$

Sintesi di un ENCODER a tre ingressi

x_2	x_1	x_0	z_1	z_0
0	0	0	0	0
1	0	0	1	1
0	1	0	1	0
0	0	1	0	1

N.B. le altre configurazioni sono per ipotesi impossibili

$z_1 = x_2 x_1 x_0' + x_2' x_1 x_0'$
 $z_0 = x_2 x_1 x_0' + x_2' x_1' x_0$

Sintesi canonica del Full Adder

$S = r'. a'. b + r'. a. b' + r. a'. b' + r. a. b$
 $R = r'. a. b + r. a'. b + r. a. b' + r. a. b$

Sintesi del trascodificatore da binario a 1 su N

Esempio: Trascodifica 2:4

B	A	U_0	U_1	U_2	U_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

$U_0 = B'. A'$
 $U_1 = B'. A$
 $U_2 = B. A'$
 $U_3 = B. A$

Manipolazione algebrica

A	I_0	I_1	U
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

I^a forma canonica:
 $U = A'. I_0. I_1' + A'. I_0. I_1 + A. I_0'. I_1 + A. I_0. I_1$

4 AND a 3 ingressi e 1 OR a 4 ingressi

forme equivalenti ottenute per "manipolazione":
 $= A'. I_0. (I_1' + I_1) + A. (I_0' + I_0). I_1$
 $= A'. I_0. 1 + A. 1. I_1$
 $= A'. I_0 + A. I_1$

Il Selettore a quattro vie

MUX a 4 vie (espressione SP)
 $U = A_1'. A_0'. I_0 + A_1'. A_0. I_1 + A_1. A_0'. I_2 + A_1. A_0. I_3$

Notazioni simboliche

Notazioni simboliche per le espressioni canoniche

i	r a b	R	S
0	000	0	0
1	001	0	1
2	010	0	1
3	011	1	0
4	100	0	1
5	101	1	0
6	110	1	0
7	111	1	1

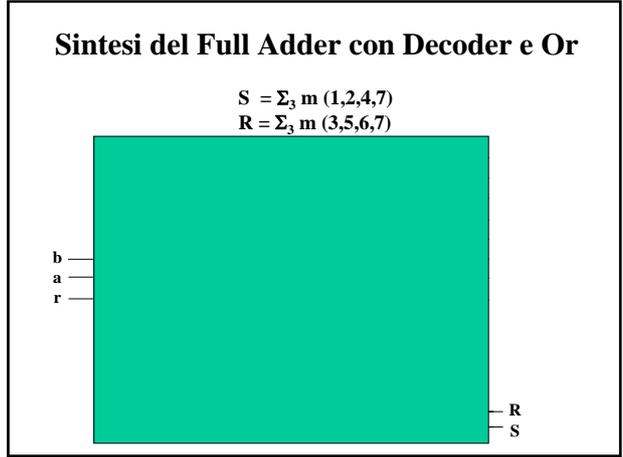
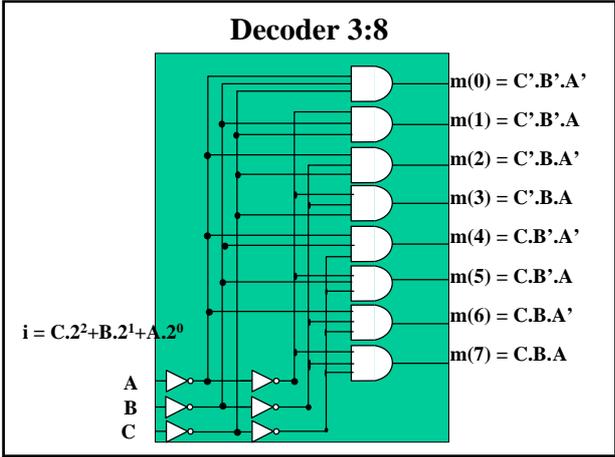
$S(r,a,b) = \Sigma_3 m(1,2,4,7)$

$S(r,a,b) = \Pi_3 M(0,3,5,6)$

$R(r,a,b) = \Sigma_3 m(3,5,6,7)$

$R(r,a,b) = \Pi_3 M(0,1,2,4)$

- **m(i)** : mintermine di n bit che assume il valore 1 solo per la n-pla di valori delle variabili corrispondente all'indice i
- **M(i)** : maxtermine di n bit che assume il valore 0 solo per la n-pla di valori delle variabili corrispondente all'indice i



Espressioni generali

Teoremi di espansione (o di Shannon)

T8) $E(x_1, x_2, \dots, x_{n-1}, x_n) = x_n' \cdot E(x_1, x_2, \dots, x_{n-1}, 0) + x_n \cdot E(x_1, x_2, \dots, x_{n-1}, 1)$

T9) $E(x_1, x_2, \dots, x_{n-1}, x_n) = (x_n + E(x_1, x_2, \dots, x_{n-1}, 0)) \cdot (x_n' + E(x_1, x_2, \dots, x_{n-1}, 1))$

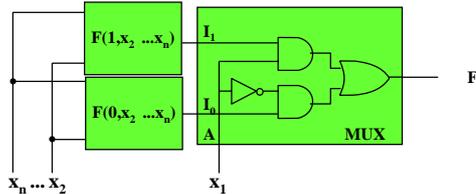
Esempio:

$E = x_1 + x_2 \cdot x_3'$

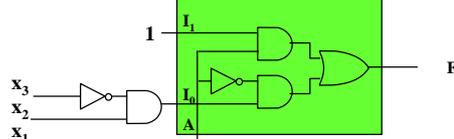
$= x_1' \cdot (0 + x_2 \cdot x_3') + x_1 \cdot (1 + x_2 \cdot x_3')$

$= (x_1 + (0 + x_2 \cdot x_3')) \cdot (x_1' + (1 + x_2 \cdot x_3'))$

Mux e teoremi di espansione



Esempio : $x_1 + x_2 x_3' = x_1' (0 + x_2 x_3') + x_1 (1 + x_2 x_3')$



Applicazione iterata dei teoremi di espansione

$$\begin{aligned}
 E(x_1 x_2 x_3) &= x_1 + x_2 x_3' \\
 &= x_1'(0 + x_2 x_3') + x_1(1 + x_2 x_3') \\
 &= x_1' x_2'(0 + 0 \cdot x_3') + x_1' x_2(0 + 1 \cdot x_3') + x_1 x_2'(1 + 0 \cdot x_3') + x_1 x_2(1 + 1 \cdot x_3') \\
 &= x_1' x_2' x_3'(0 + 0 \cdot 0) + x_1' x_2' x_3(0 + 0 \cdot 1) + x_1' x_2 x_3'(0 + 1 \cdot 0) + x_1' x_2 x_3(0 + 1 \cdot 1) \\
 &\quad + x_1 x_2' x_3'(1 + 0 \cdot 0) + x_1 x_2' x_3(1 + 0 \cdot 1) + x_1 x_2 x_3'(1 + 1 \cdot 0) + x_1 x_2 x_3(1 + 1 \cdot 1) \\
 &= m(0).E(0) + m(1).E(1) + m(2).E(2) + m(3).E(3) + m(4).E(4) + m(5).E(5) + m(6).E(6) + m(7).E(7)
 \end{aligned}$$

Espressioni generali

T10 e T11)- Ogni funzione è descritta da una espressione in cui compaiono o tutti i mintermini o tutti i maxtermini:

$$F(x_1, x_2, \dots, x_{n-1}, x_n) = \sum_{i=0}^{2^n-1} m(i) \cdot F(i) \quad (SP)$$

$$F(x_1, x_2, \dots, x_{n-1}, x_n) = \prod_{i=0}^{2^n-1} (M(i) + F(i)) \quad (PS)$$

Caso SP

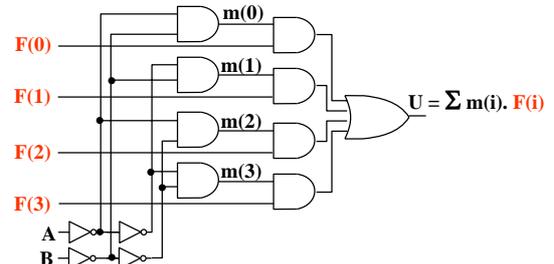
m(i) : mintermine di n bit
F(i): **valore** dalla funzione per la n-pla di valori delle variabili per cui m(i)=1

Caso PS

M(i) : maxtermini di n bit
F(i): **valore** dalla funzione per la n-pla di valori delle variabili per cui M(i)=0

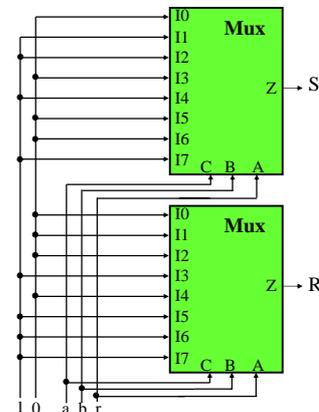
La rete combinatoria programmabile

Non conviene realizzare separatamente tutte le funzioni di n variabili e scegliere di volta in volta quella che serve. E' molto più utile fare un'unica rete che le realizza tutte con una semplice "programmazione" dei suoi ingressi.



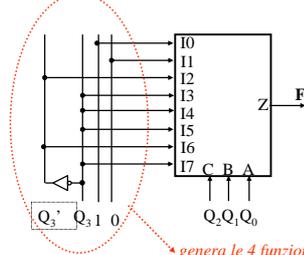
Sintesi di un full-adder con MUX

a	b	r	S	R
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



Pag.76 - Funzioni di n variabili e MUX con n-1 bit d'indirizzo

Q ₃	Q ₂ Q ₁ Q ₀	000	001	010	011	100	101	110	111
0		1	0	1	0	0	0	1	0
1		1	0	0	1	1	1	0	1

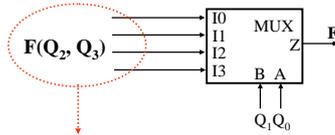


- F(0,0,0,Q₃) = 1
- F(0,0,1,Q₃) = 0
- F(0,1,0,Q₃) = Q₃'
- F(0,1,1,Q₃) = Q₃
- F(1,0,0,Q₃) = Q₃
- F(1,0,1,Q₃) = Q₃
- F(1,1,0,Q₃) = Q₃'
- F(1,1,1,Q₃) = Q₃

NUOVA! - Funzioni di n variabili e MUX con n-2 bit d'indirizzo

	Q_1, Q_0				
	00	01	10	11	
Q_2, Q_3	00	1	0	1	0
	10	1	0	0	1
	01	0	0	1	0
	11	1	1	0	1

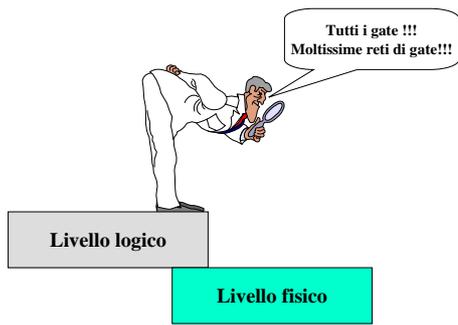
$F(0,0,Q_2,Q_3) =$
 $F(1,0,Q_2,Q_3) =$
 $F(0,1,Q_2,Q_3) =$
 $F(1,1,Q_2,Q_3) =$



16 funzioni di due variabili!

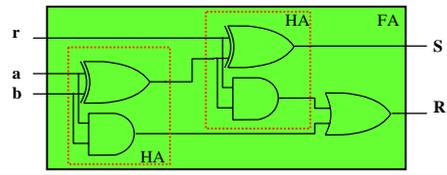


Famiglie di circuiti logici integrati



Full Adder con AND, OR e EX-OR

$S = r'. a'. b + r'. a . b' + r . a'. b' + r . a . b$
 $R = r'. a . b + r . a'. b + r . a . b' + r . a . b$
 manipolazione algebrica:
 $S = r'. (a'. b + a . b') + r . (a'. b' + a . b)$
 $S = r'. (a \oplus b) + r . (a \oplus b)'$
 $S = r \oplus (a \oplus b)$
 $R = (r' + r) . a . b + r . (a'. b + a . b')$
 $R = a . b + r . (a \oplus b)$

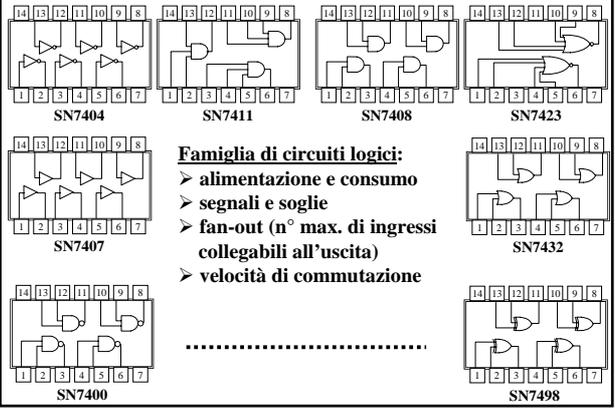


Confronto tra due numeri di n bit

$CFR = (a_0'. b_0' + a_0 . b_0) . (a_1'. b_1' + a_1 . b_1)$

CELLA:
 2 AND a due ingressi
 1 OR a due ingressi
 2 NOT
CELLA:
 1 EX-NOR a due ingressi

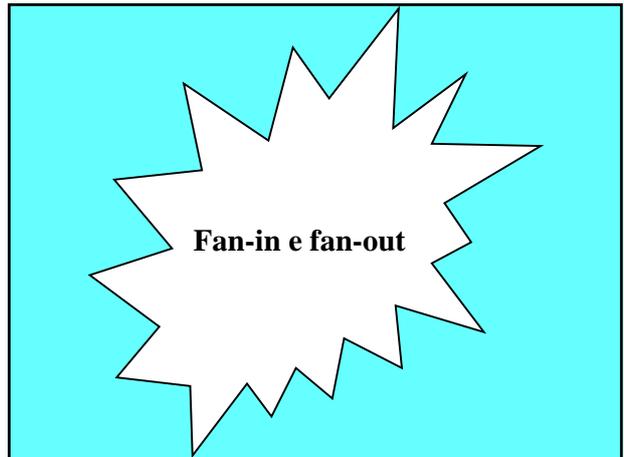
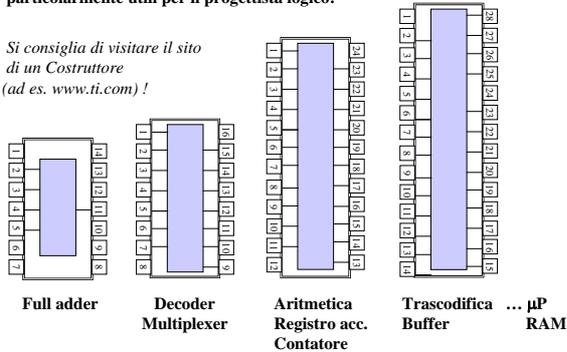
Famiglie di gate (TTL SSI -1968/74)



Circuiti combinatori MSI e LSI

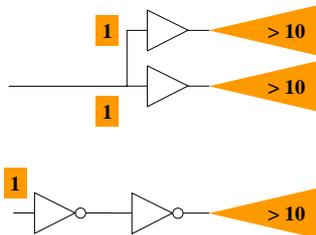
Sono disponibili come parti elementari anche reti di gate particolarmente utili per il progettista logico:

Si consiglia di visitare il sito di un Costruttore (ad es. www.it.com)!

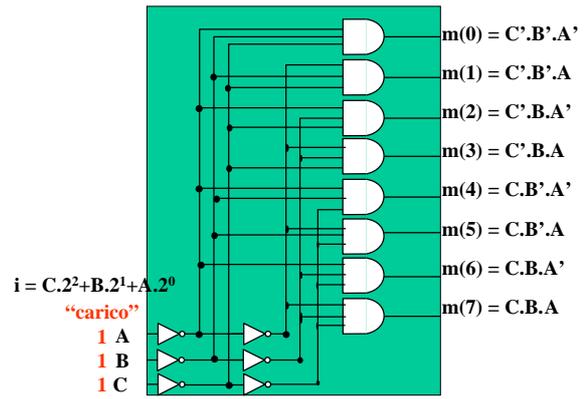


Effetto di carico: uso di Buffer e Not

Fan-out
 "l'uscita di un gate ha un numero massimo di ingressi di altri gate a cui può essere collegata"



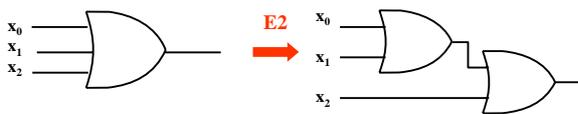
Decoder 3:8



And e Or: proprietà associativa

Fan-in

Gate con un massimo di otto ingressi



Parità con EX-OR (1)

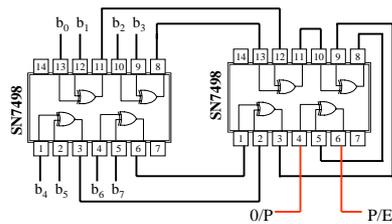
Fan-in

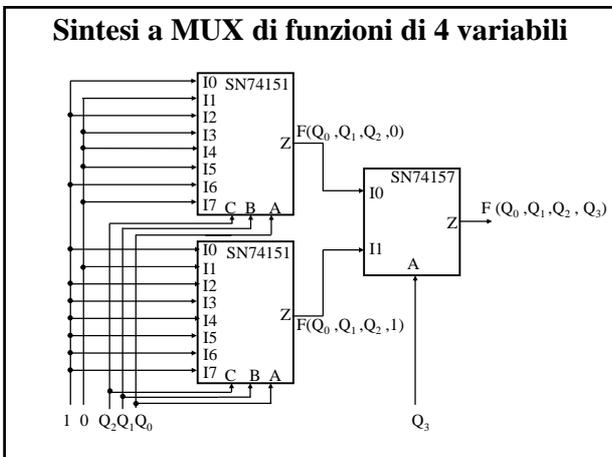
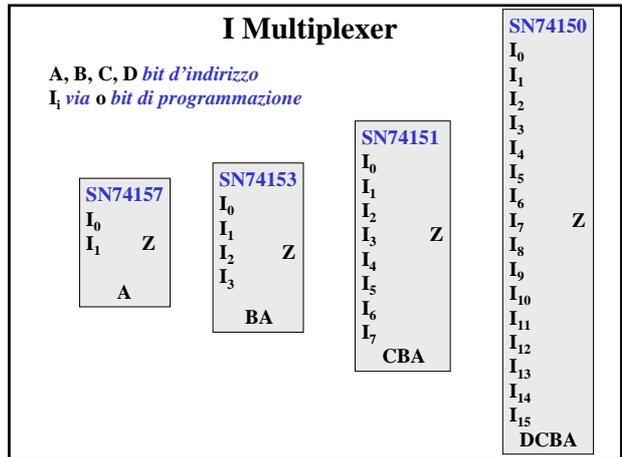
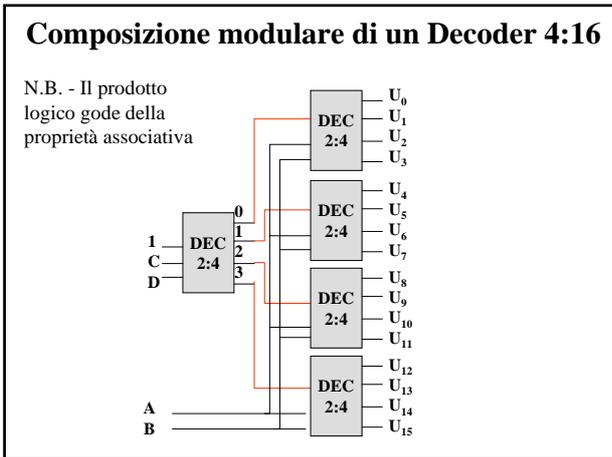
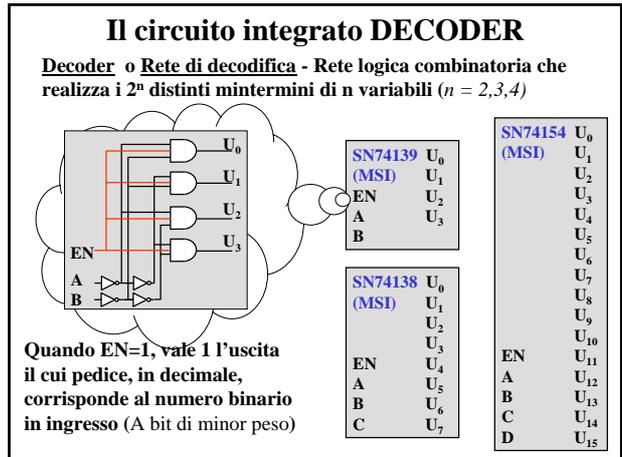
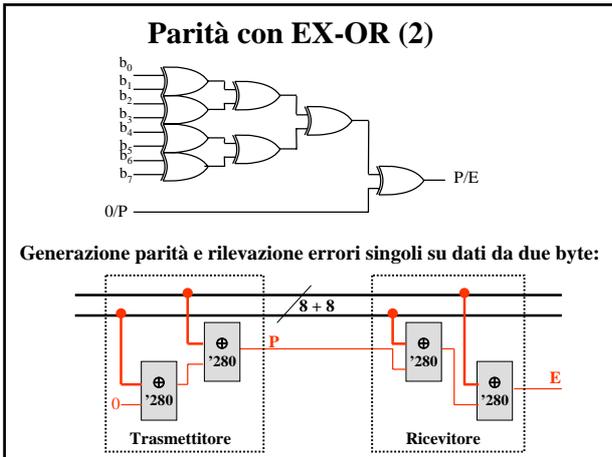
$$P = b_0 \oplus b_1 \oplus b_2 \oplus b_3 \oplus \dots \oplus b_7$$

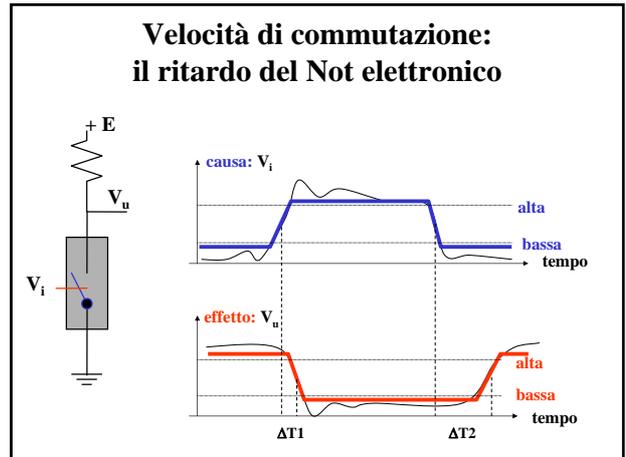
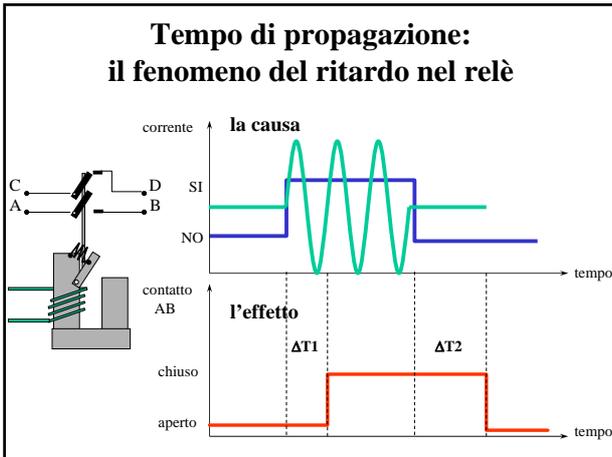
N.B. L'operazione di somma modulo due è associativa

$$P = ((b_0 \oplus b_1) \oplus (b_2 \oplus b_3)) \oplus (\dots \oplus b_7)$$

$$E = P \oplus (((b_0 \oplus b_1) \oplus (b_2 \oplus b_3)) \oplus (\dots \oplus b_7))$$

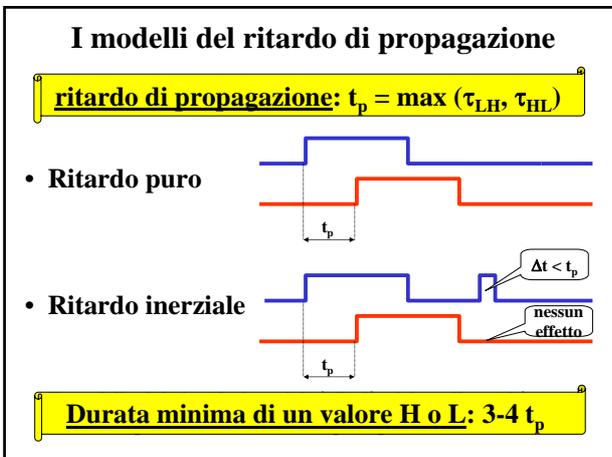
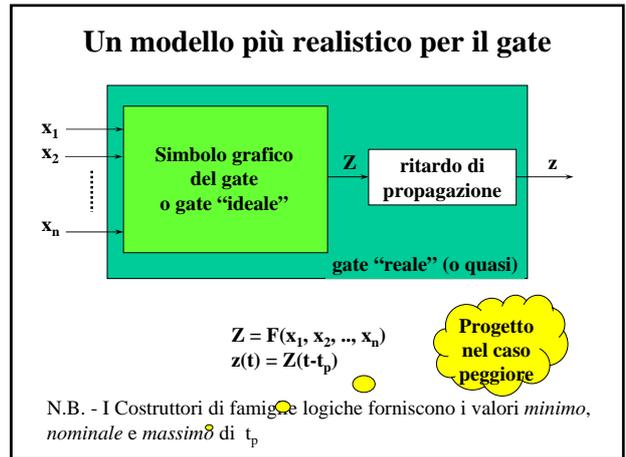






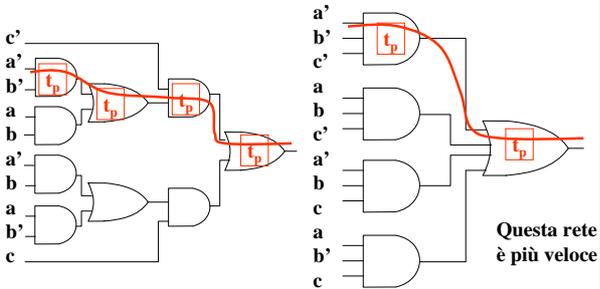
Il ritardo sui fronti

- Il **ritardo sui fronti di salita** (τ_{LH}) e di discesa (τ_{HL}) è presente in ogni tipo di gate e varia in modo notevole da dispositivo a dispositivo.
- A causa della marcata differenza dei due valori, la durata di una situazione H o L in ingresso ad un gate è diversa dalla corrispondente situazione in uscita.
- A causa della “inerzia” del gate, un segnale di ingresso “impulsivo” e “troppo stretto” può **non** essere avvertito in uscita.



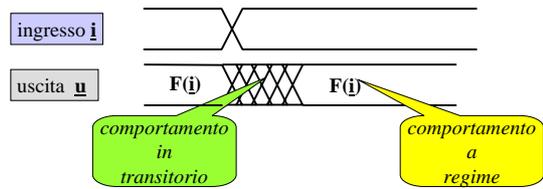
Velocità e lunghezza dei percorsi

$$(a'.b'+a.b).c'+(a'.b+a.b').c = a'.b'.c' + a.b.c' + a'.b.c + a.b'.c$$

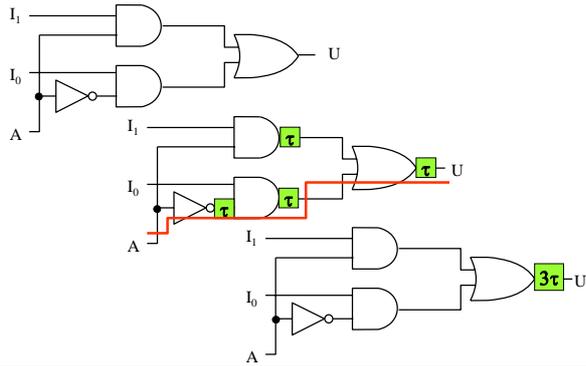


Comportamento a regime e in transitorio dei circuiti combinatori

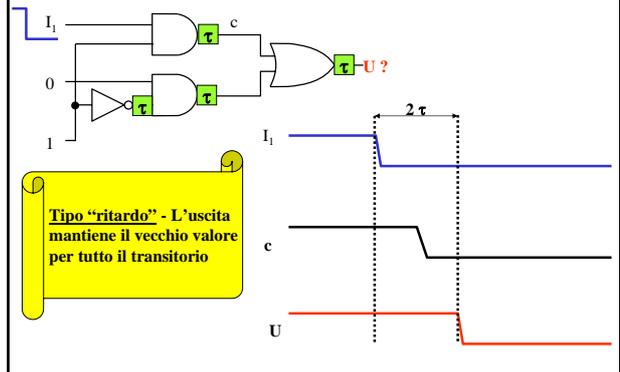
I nuovi valori dei segnali di ingresso di una rete combinatoria devono propagarsi all'interno della struttura prima di riuscire ad imporre al segnale d'uscita il valore che ad essi deve corrispondere. Ciò determina un comportamento in transitorio, che in generale sarà diverso da quello a regime.



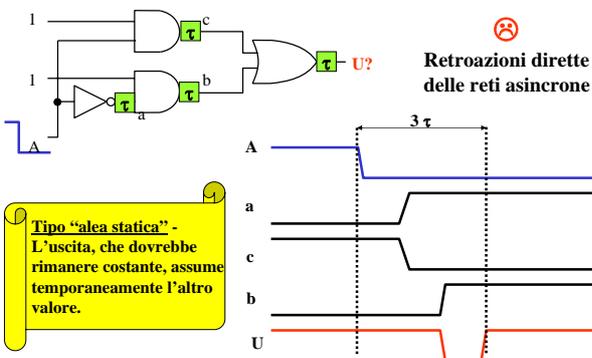
Stima della durata del transitorio (metodo del caso peggiore)



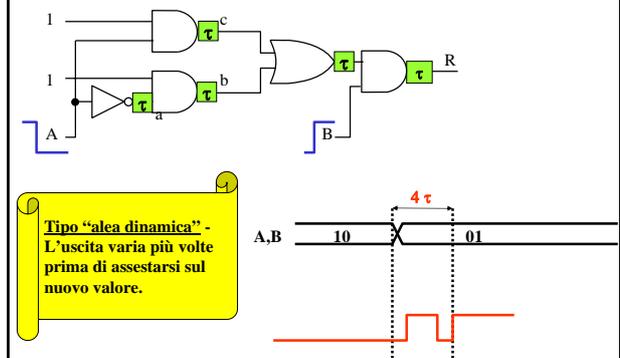
Tipi di transitorio: il ritardo

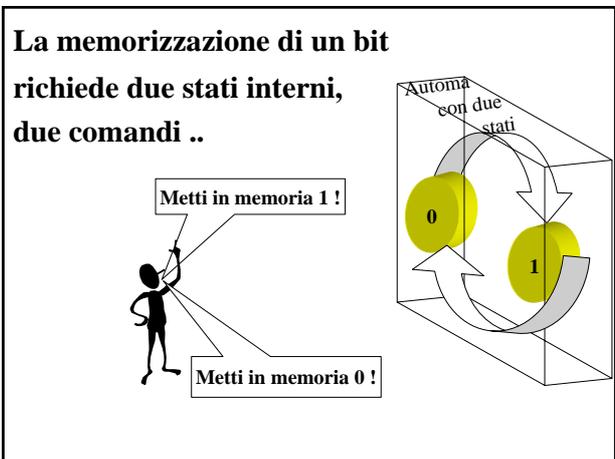
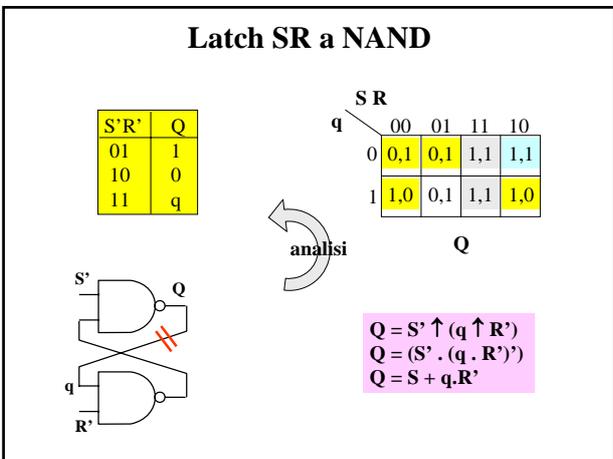
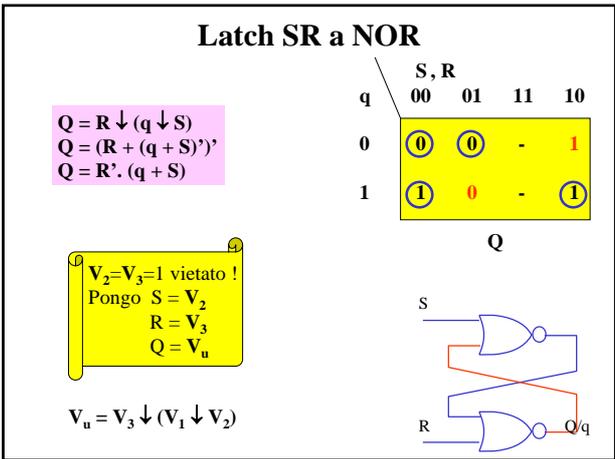
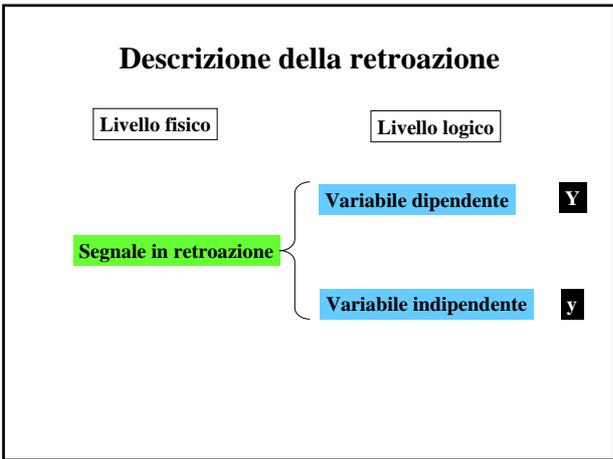
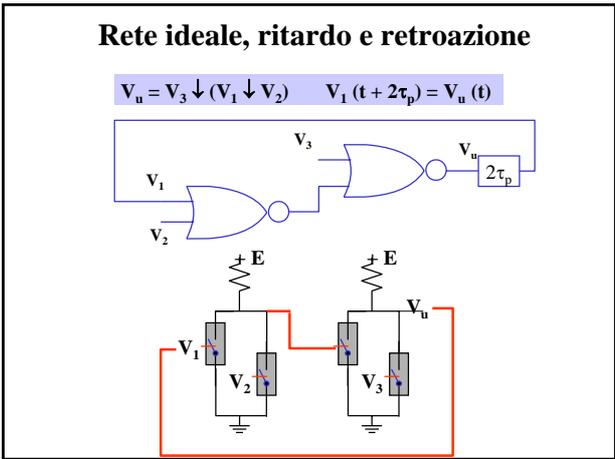


Tipi di transitorio: l'alea statica

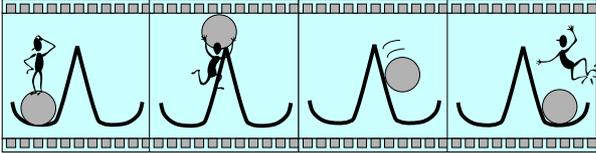


Tipi di transitorio: l'alea dinamica

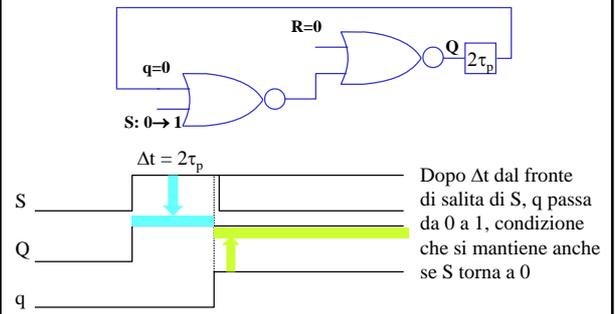




... e un po' di fatica !



Cosa occorre per scrivere un 1 (o uno 0)?



La durata minima di un comando di set/reset è spesso indicata con la denominazione di *tempo di set-up* del latch