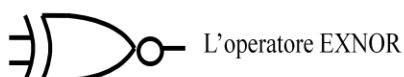
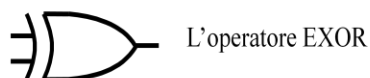
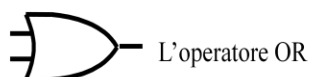
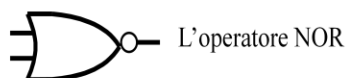
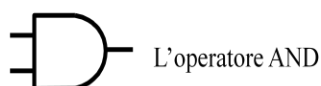
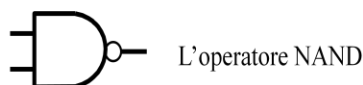
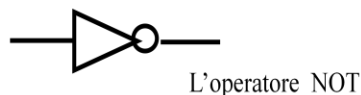


Cap. 3 – Reti combinatorie: analisi e sintesi operatori logici e porte logiche

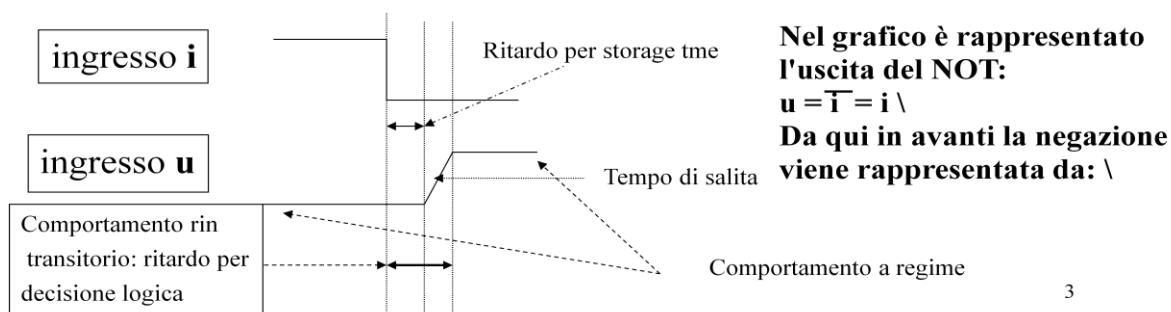
3.1 LE PORTE LOGICHE E GLI OPERATORI ELEMENTARI



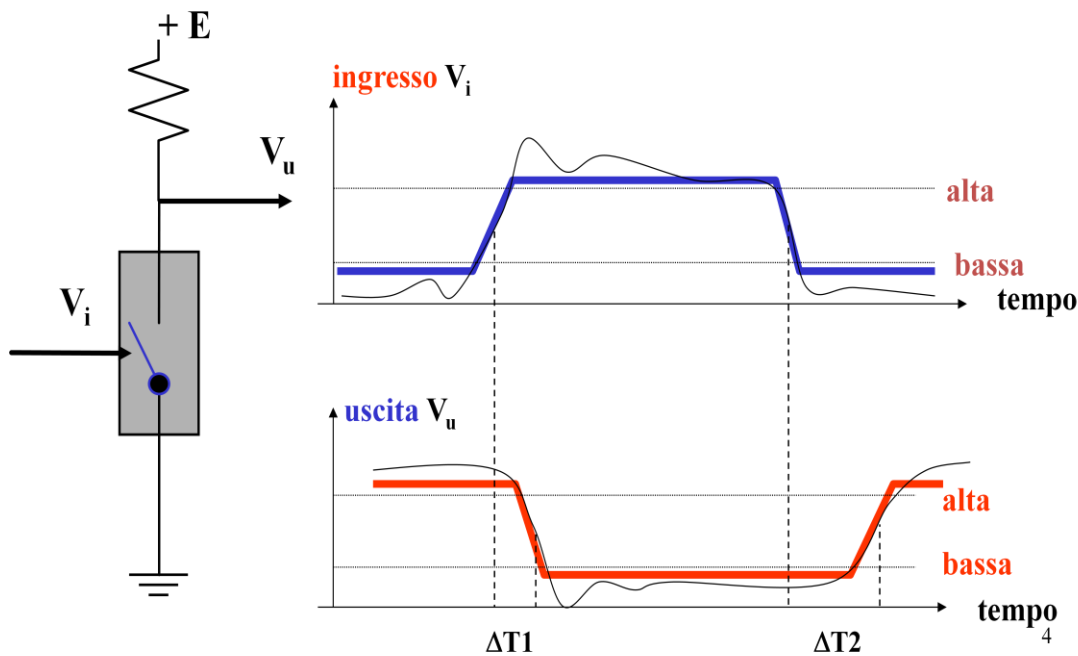
3.2 COMPORTAMENTO A REGIME E IN TRANSITORIO DEI CIRCUITI COMBINATORI

I nuovi valori dei segnali di ingresso di una rete combinatoria devono propagarsi all'interno della rete prima di 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:

- il comportamento a regime è quello previsto dal modello,
- il comportamento in transitorio si compone a sua volta di due parti:
 - o ritardo per storage time
 - o tempo di salita o di discesa



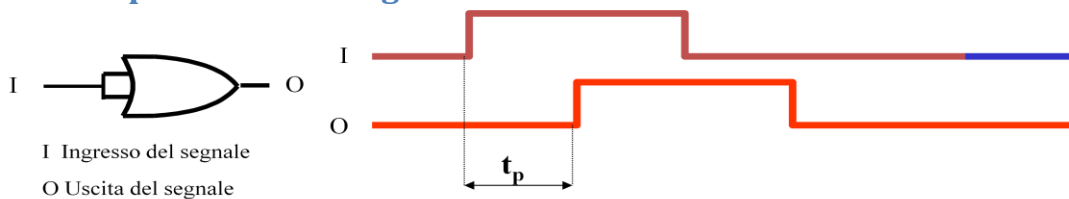
Velocità di commutazione: il ritardo del NOT elettronico



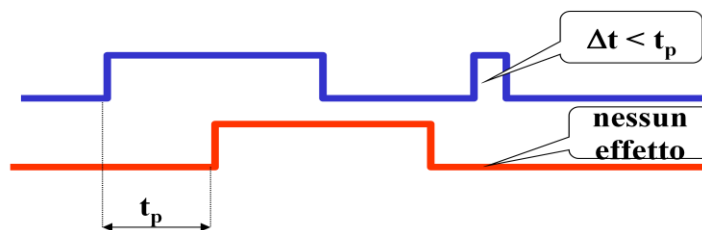
Il ritardo sui fronti:

- Il ritardo sui fronti di salita (t_{LH}) e di discesa (t_{HL}) è presente in ogni tipo di gate e varia in modo notevole da dispositivo a dispositivo.
- A causa della 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.

Il ritardo per decisione logica



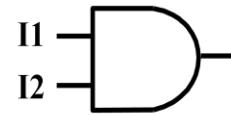
Se un segnale ha durata minore di t_p esso viene ignorato in uscita



La relazione ingresso/uscita e il simbolo grafico degli operatori logici AND e OR

Operatore logico AND

L'uscita è 1 quando tutti gli ingressi sono 1

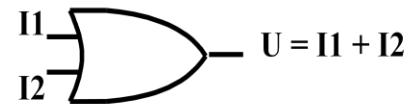


Operatore logico OR

L'uscita è 1 quando almeno un ingresso è 1

I1	I2	U
0	0	0
0	1	1
1	0	1
1	1	1

Tabella della verità



N.B. i due operatori possono avere più di due ingressi

La relazione ingresso/uscita e il simbolo grafico di altri operatori logici

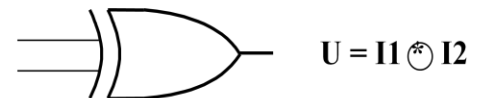
Operatore logico EXOR

L'uscita è 1 quando i due ingressi sono uguali

I1	I2	U
0	0	0
0	1	1
1	0	1
1	1	0

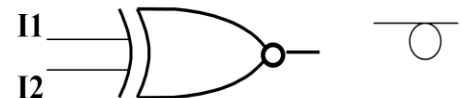
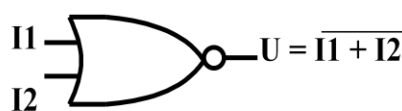
Tabella della verità

Simbolo grafico



Oltre ai tre operatori visti esistono gli operatori:

NOR, NAND e EXNOR la cui uscita è negata, i loro simboli sono:

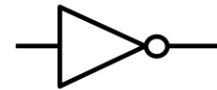


La relazione ingresso/uscita e il simbolo grafico dell'operatore NOT

Operatore logico NOT

I	U
0	1
1	0

Tabella della verità



Simbolo grafico

Gli altri operatori logici fondamentali

- L'operatore NAND ha l'uscita negata rispetto all'operatore AND
- L'operatore NOR ha l'uscita negata rispetto all'operatore OR
- L'operatore EXOR ha uscita 1 quando i due ingressi sono diversi
- L'operatore EXNOR ha uscita 1 quando i due ingressi sono uguali

N.B. Mentre gli operatori AND, OR, NOR e NAND possono avere un numero d'ingressi maggiore di 2 l'EXOR e l'EXNOR hanno solo due ingressi.

3.3 ANALISI E SINTESI DI RETI COMBINATORIE

Equivalenze notevoli dell'algebra di commutazione

Proprietà della somma e del prodotto logico:

T4) <i>commutativa</i>	$x + y$	=	$y + x$
	$x \cdot y$	=	$y \cdot x$
T5) <i>associativa</i>	$(x + y) + z$	=	$x + y + z$
	$(x \cdot y) \cdot z$	=	$x \cdot y \cdot z$
T6) <i>distributiva</i>	$(x \cdot y) + (x \cdot z)$	=	$x \cdot (y + z)$
	$(x + y) \cdot (x + z)$	=	$x + (y \cdot z)$
T7) <i>idempotenza</i>	$x + x$	=	x
	$x \cdot x$	=	x
T8) <i>identità</i>	$x + 0$	=	x
	$x \cdot 1$	=	x
T9) <i>limite</i>	$x + 1$	=	1

$$x \cdot 0 = 0$$

Altre equivalenze notevoli dell'algebra di commutazione

$$T10) \text{ involuzione} \quad (x \setminus) \setminus = x$$

$$T11) \text{ limitazione} \quad x + x \setminus = 1$$

$$x \cdot x \setminus = 0$$

$$T12) \text{ combinazione} \quad xy + xy \setminus = x$$

$$(x+y) \cdot (x+y \setminus) = x$$

$$\text{legge di De Morgan} \quad (x + y) \setminus = x \setminus \cdot y \setminus$$

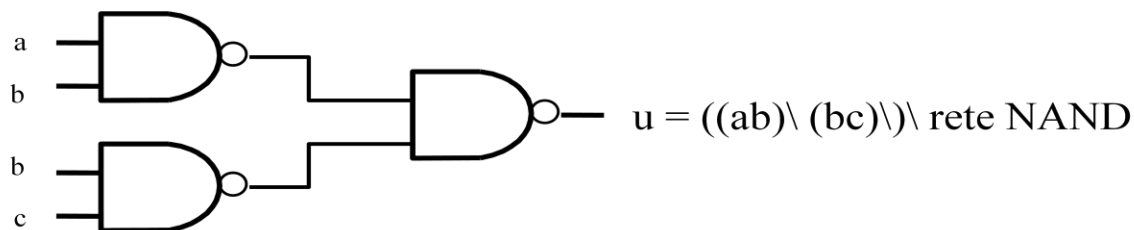
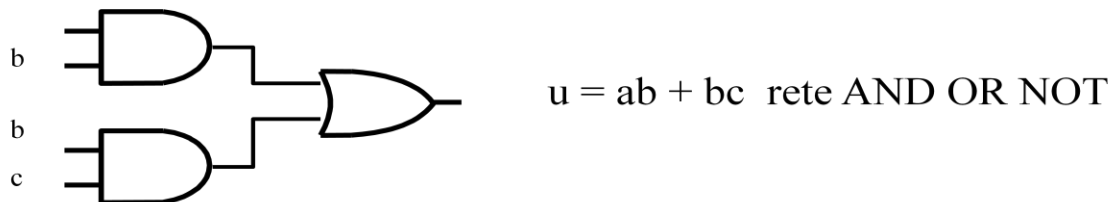
$$\text{legge di De Morgan} \quad (x \cdot y) \setminus = x \setminus + y \setminus$$

Espressioni SOP = RETI AND OR NOT E RETI NAND

Le espressioni algebriche che abbiamo visto: espressioni SOP e POS sono realizzabili con reti AND, OR e NOT. Accanto a queste soluzioni possiamo realizzare reti:

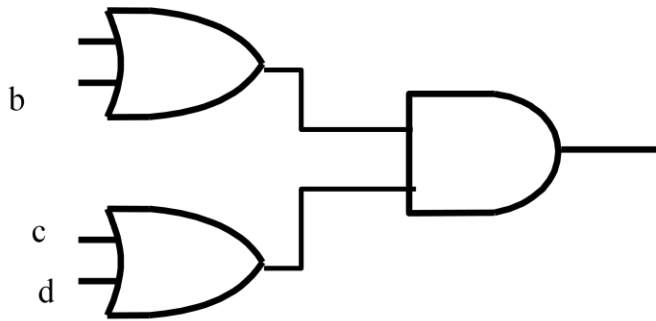
- solo con porte NOR
- solo con porte NAND

Prendiamo una espressione SOP: $u = ab + cd = (ab + cd) \setminus \setminus = ((ab) \setminus (cd) \setminus) \setminus =$ rete di soli NAND



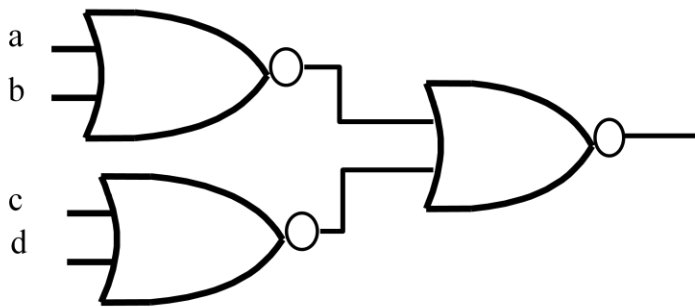
Espressioni POS = RETI AND OR NOT E RETI NOR

$$u = (a + b)(c + d) = ((a + b)(c + d))\backslash\backslash = ((a + b)\backslash + (c + d)\backslash)\backslash$$



Rete AND OR NOT

$$u = (a + b)(c + d)$$

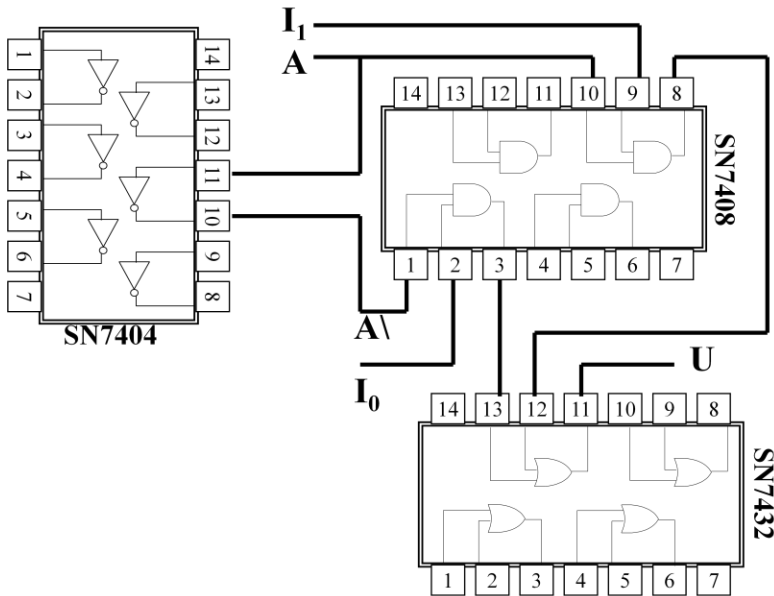


Rete NOR

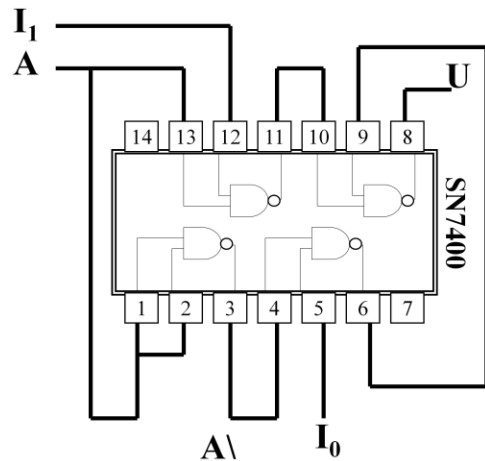
$$u = ((a + b)\backslash + (b + c)\backslash)\backslash$$

Sintesi con rete AND, OR, NOT E RETE CON SOLI NAND

$$U = A \setminus I_0 + A \cdot I_1$$



$$U = \overline{(A \setminus I_0)} \cdot \overline{(A \cdot I_1)}$$



La realizzazione con rete NAND richiede un numero di circuiti integrati SSI molto minore, lo stesso dicasi se avessimo realizzato il circuito con sole porte NOR.

Regole di priorità nella valutazione

- Si ricordi che, in assenza di parentesi valgono le seguenti regole:
 - L'operazione di complementazione è prioritaria rispetto a prodotto e somma
 - L'operazione di prodotto è prioritaria rispetto alla somma e non è obbligatorio racchiuderla tra parentesi.

Rappresentazione e realizzazione di una rete combinatoria

Una rete combinatoria può essere descritta in tre diverse maniere:

- tabella della verità
- espressione algebrica
- mappa di Karnaugh

La realizzazione circuitale di una qualsiasi rete combinatoria può essere realizzata con i seguenti operatori:

- AND, OR, NOT
- solo NOR
- solo NAND

Tabella della verità del full adder somma di due cifre binarie tenuto conto del riporto

S = A plus B

Espressioni SOP non ottimizzate: somme di mintermini:

r a b	R	S
0 0 0	0	0
0 0 1	0	1
0 1 0	0	1
0 1 1	1	0
1 0 0	0	1
1 0 1	1	0
1 1 0	1	0
1 1 1	1	1

$$S = r\backslash a\backslash b + r\backslash a \cdot b\backslash + r \cdot a\backslash b\backslash + r \cdot a \cdot b$$

$$R = r\backslash a \cdot b + r \cdot a\backslash b + r \cdot a \cdot b\backslash + r \cdot a \cdot b$$

Rete realizzabile con AND OR e NOT

Applicando la legge di MORGAN

$$S = ((r\backslash a\backslash b)\backslash (r\backslash a \cdot b\backslash) \cdot (r \cdot a\backslash b\backslash) \cdot (r \cdot a \cdot b)\backslash)$$

$$R = ((r\backslash a \cdot b) \cdot (r \cdot a\backslash b)\backslash (r \cdot a \cdot b\backslash) \cdot (r \cdot a \cdot b)\backslash)$$

Rete realizzabile con solo porte NAND

Tabella della verità del full adder somma di due cifre binarie tenuto conto del riporto

S = A plus B

Espressioni POS non ottimizzate: prodotti di maxitermini:

r	a	b	R	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

$$S = (a + b + c) (a + b\backslash + c\backslash) (a\backslash + b + c\backslash) (a\backslash + b\backslash + c)$$

$$R = (a + b + c) (a + b + c\backslash) (a + b\backslash + c) (a\backslash + b + c)$$

Rete realizzabile con AND OR e NOT

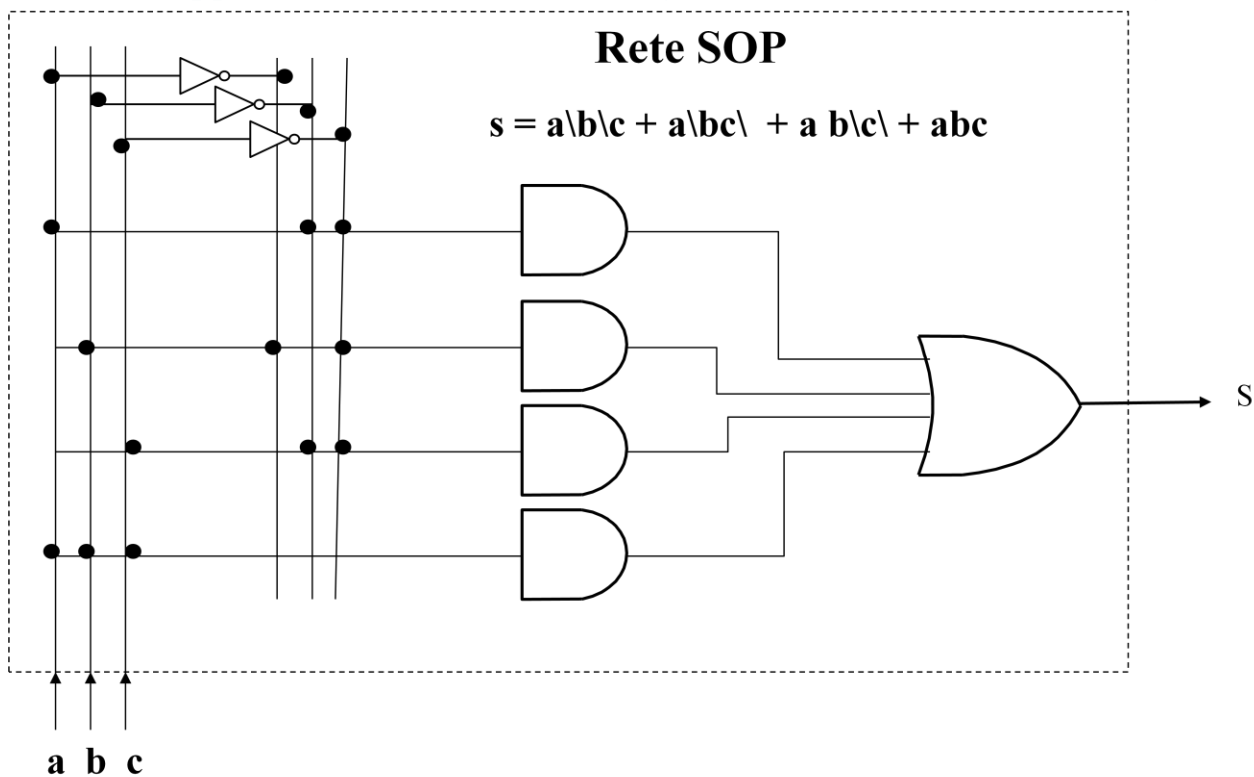
Applicando la legge di MORGAN

$$S = ((a + b + c)\backslash + (a + b\backslash + c\backslash)\backslash + (a\backslash + b + c\backslash)\backslash + (a\backslash + b\backslash + c)\backslash)\backslash$$

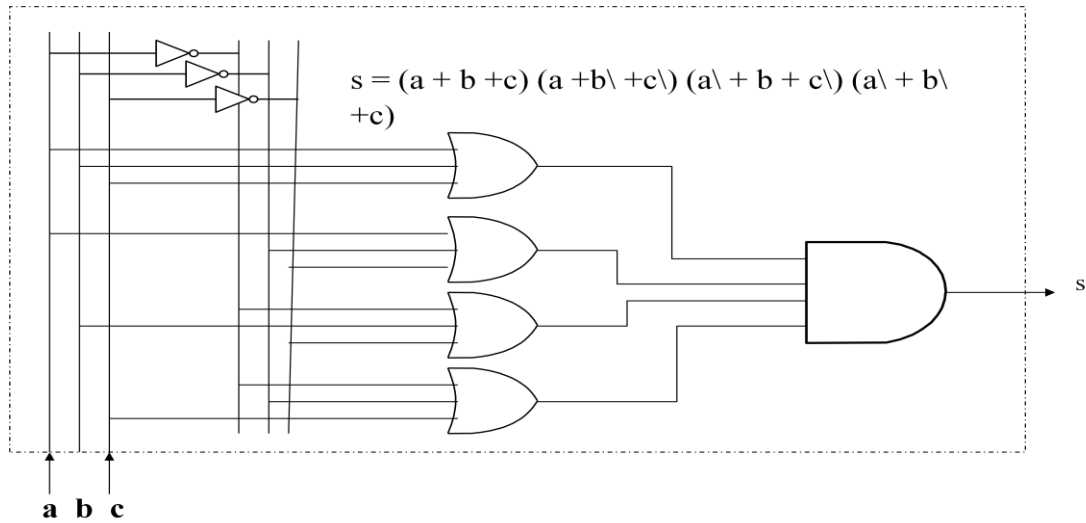
$$R = ((a + b + c)\backslash + (a + b + c\backslash)\backslash + (a + b\backslash + c)\backslash + (a\backslash + b + c)\backslash)\backslash$$

Rete realizzabile con solo porte NOR

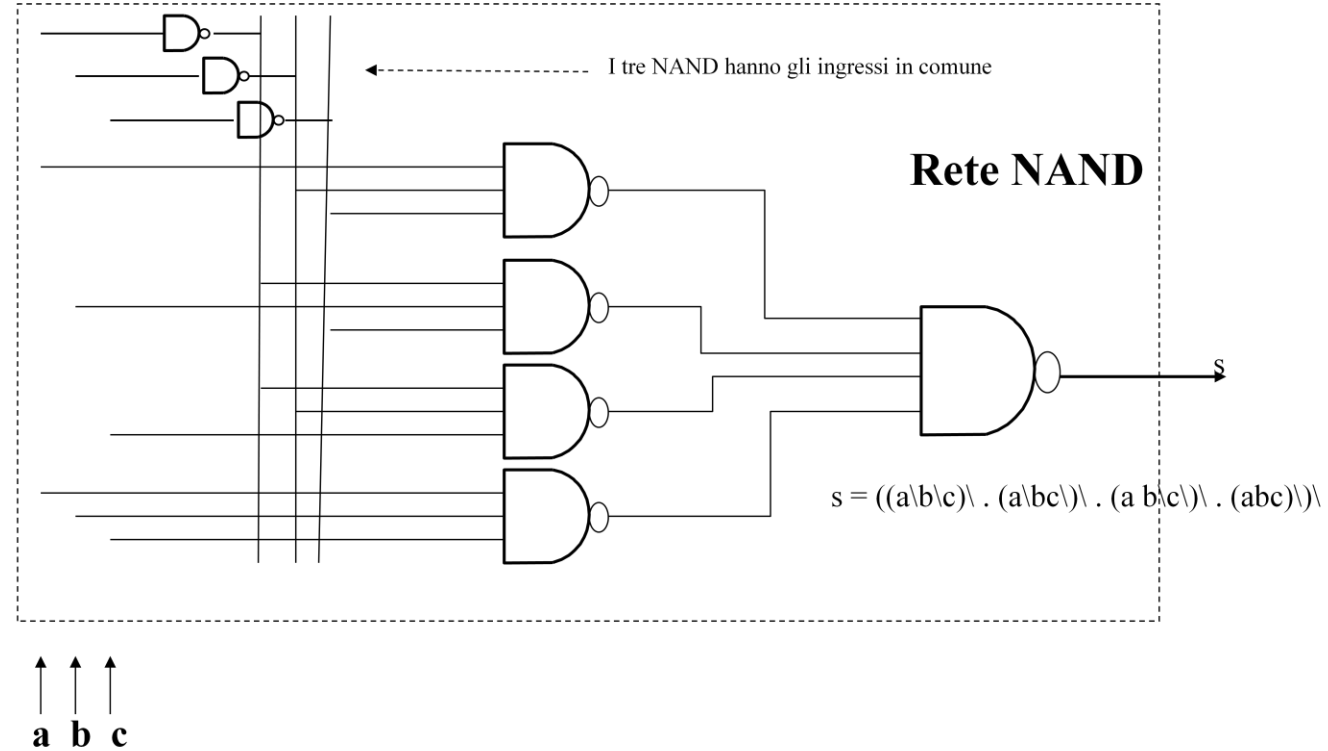
Corrente rappresentazione di una rete logica: rete combinatoria a tre ingressi e una uscita



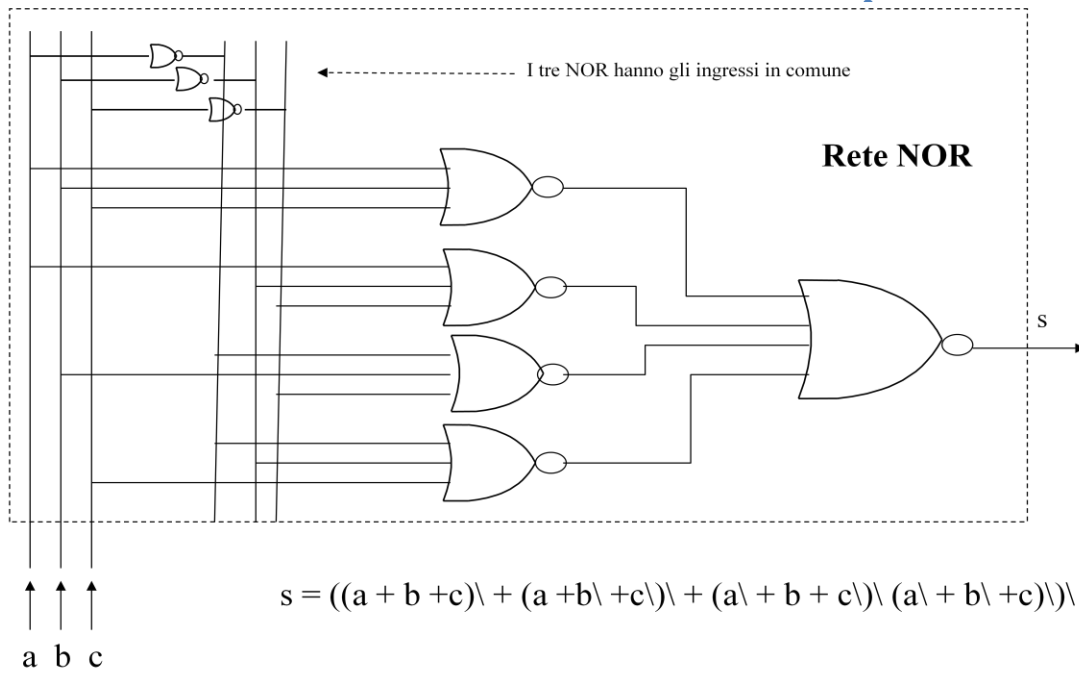
Realizzazione circuitale del bit somma aritmetica di tre cifre: espressione POS con porte AND, OR e NOT



Realizzazione circuitale del bit somma: con porte NAND da espressione SOP



Realizzazione circuitale del bit somma: rete NOR da espressione POS

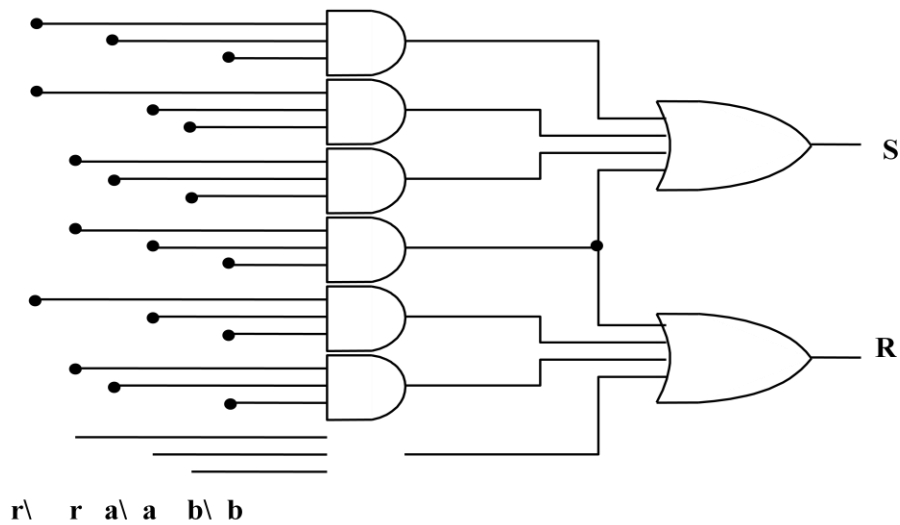


Sintesi canonica del Full Adder

$$S = r\backslash . a\backslash . b + r\backslash . a . b\backslash + r . a\backslash . b\backslash + r . a . b$$

$$R = r\backslash . a . b + r . a\backslash . b + r . a . b\backslash + r . a . b$$

Espressioni SOP
non ottimizzata



3.4 DESCRIZIONE DI UNA FUNZIONE CON LE MAPPE DI KARNAUGH

Mappa di Karnaugh - Rappresentazione bidimensionale della tabella della verità di una funzione di 2 o 4 variabili, i cui valori sono stati elencati sui bordi in maniera che due configurazioni consecutive differiscano per il valore di un solo bit. Il valore della funzione è riportato nella cella corrispondente

		b	
		0	1
a	0	0	1
	1	1	1

Somma

		cd			
		00	01	11	10
ab	00	0	1	0	1
	01	1	0	1	0
	11	0	1	0	1
	10	1	0	1	0

Parità pari su 4 bit

Scopo di queste rappresentazione è quello di individuare in maniera immediata prodotti che possono dare luogo a semplificazioni come quelle già viste.

Importante proprietà delle mappe di Karnaugh: adiacenza tra celle e ottimizzazione della espressione algebrica

Coppia di celle adiacenti su mappe di Karnaugh - Due celle le cui coordinate differiscono per un solo bit. Se in queste celle vi sono 1 i relativi due prodotti possono dare luogo a un prodotto unico con un numero minore di variabili. Sono adiacenti celle aventi un lato in comune oppure poste all'estremità di una stessa riga o colonna.

Vedi esempi seguenti ove è stata introdotta la semplificazione della funzione f:

		b	
		0	1
a	0	1	1
	1	0	0

f

$$f = a \backslash b \backslash + a \backslash b = a \backslash$$

Per maggiori
dettagli vedi
Cap 4

		bc			
		00	01	11	10
a	0	1	1	0	0
	1	1	0	0	1

f

$$f = a \backslash b \backslash c \backslash + a \backslash b \backslash c +$$

$$a b \backslash c \backslash + a b c \backslash =$$

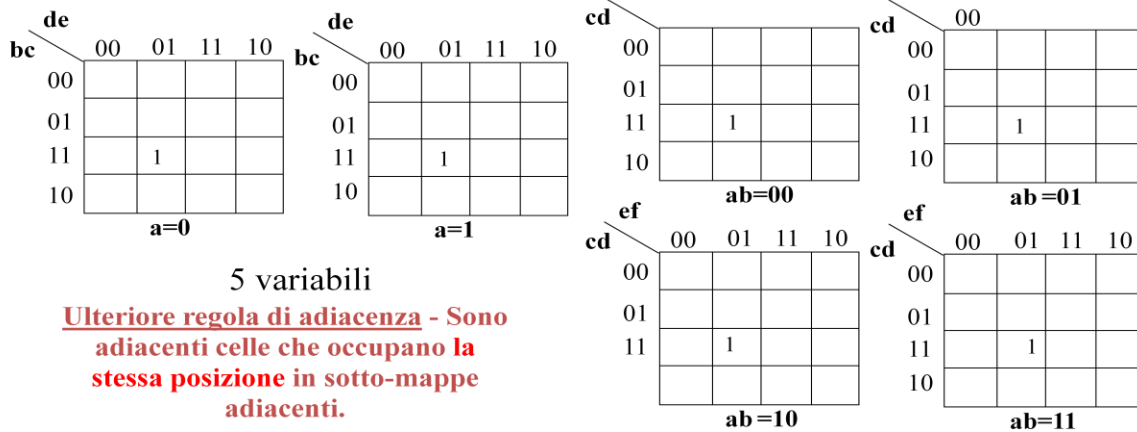
$$= a \backslash b \backslash + a c \backslash$$

		cd			
		00	01	11	10
ab	00	1	0	0	1
	01	1	0	0	1
	11	1	0	0	1
	10	1	0	0	1

f

$$f = d \backslash$$

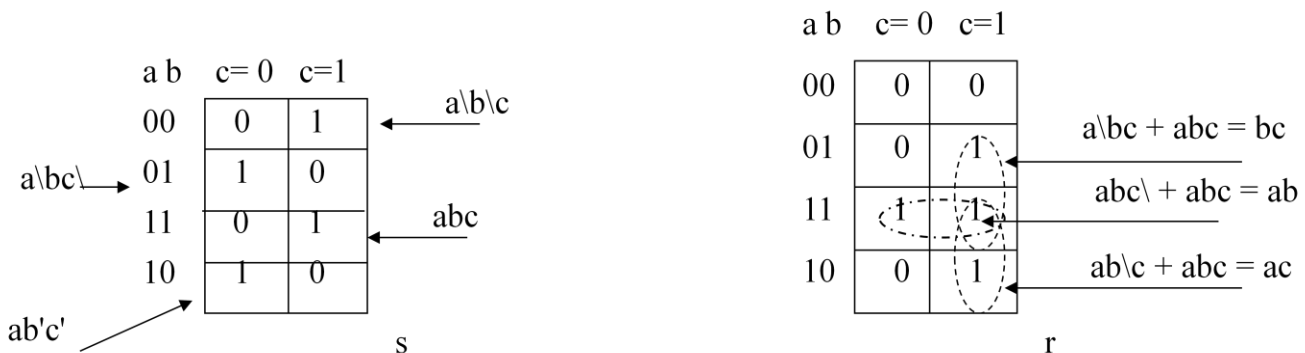
Estensione delle mappe a 5 e 6 variabili



Oggi esistono diversi SW applicativi che provvedono alla ottimizzazione della rete

Le due soluzioni circuitali precedenti, bit somma e bit riporto, non sono ottimizzate dal punto di vista del numero di porte logiche. Per vedere come una espressione SOP o POS possa essere ottimizzata dal punto di vista algebrico utilizziamo la rappresentazione della funzione con le mappe di Karnaugh.

Vediamo la rappresentazione con mappe di Karnaugh dei bit somma, s e riporto r. Nelle mappe in ciascuna cella è riportato il valore della funzione, le celle sono distribuite in maniera tale che due celle adiacenti differiscono solo per il valore di una variabile: l'individuazione di due celle adiacenti permette l'immediata individuazione di una semplificazione.

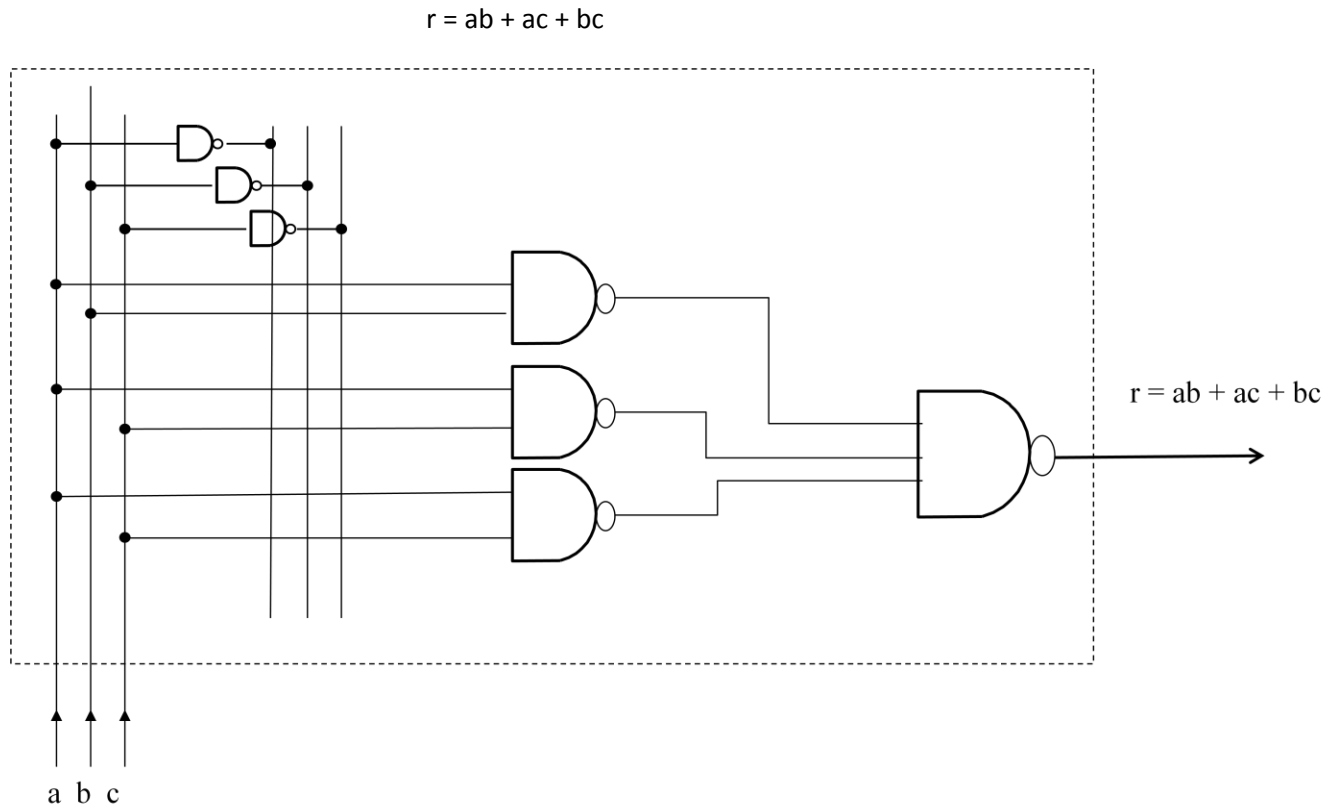


Mentre per il bit s non abbiamo celle adiacenti, per il bit r si così che la relativa espressione può essere ridotta alla somma di tre prodotti di due variabili:

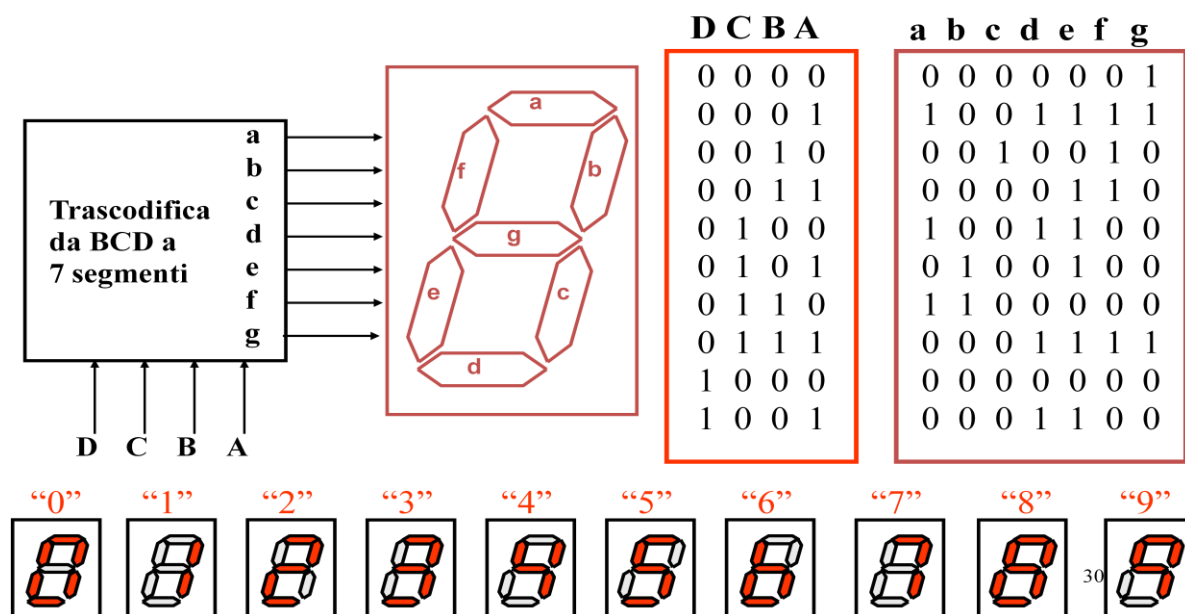
$$r = ab + ac + bc$$

Lo stesso può farsi per espressioni POS, ci occuperemo solo di espressioni SOP in quanto, come vedremo, tutte le applicazioni con PLD e PROM utilizzano solo questa rappresentazione.

Realizzazione circuitale del bit riporto: espressione SOP semplificata con porte NAND



Sintesi di un trascodificatore da BCD a 7 segmenti



Progetto della rete di costo minimo (1)

		BA			
DC		00	01	11	10
	00	0	1	0	0
	01	1	0	0	1
	11	-	-	-	-
	10	0	0	-	-

a

$$a = D \setminus C \setminus B \setminus A + CA \setminus$$

		BA			
DC		00	01	11	10
	00	0	0	0	0
	01	0	1	0	1
	11	-	-	-	-
	10	0	0	-	-

b

$$b = CB \setminus A + CBA \setminus$$

		BA			
DC		00	01	11	10
	00	0	0	0	1
	01	0	0	0	0
	11	-	-	-	-
	10	0	0	-	-

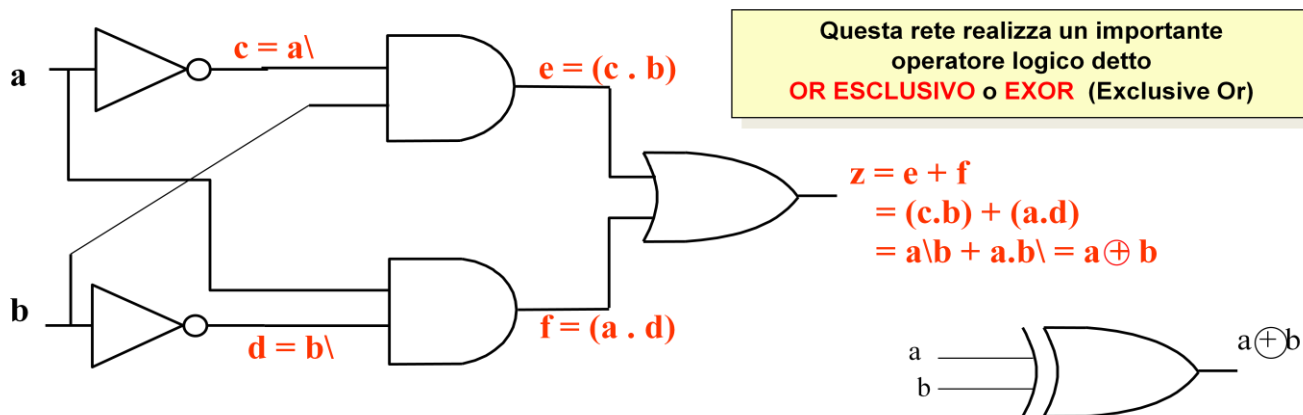
c

$$c = C \setminus BA \setminus$$

Dallo schema logico all'espressione

Per individuare l'espressione corrispondente ad un dato schema si parte dai gate che elaborano solo segnali di ingresso, si assegna un simbolo alla loro uscita e si annota a parte l'espressione.

Si procede in modo analogo con i gate i cui ingressi sono già stati denominati. Una volta individuata l'espressione del gate di uscita, vi si sostituiscono tutti i simboli con le corrispondenti espressioni.



3.5 ESPRESSIONI CANONICHE

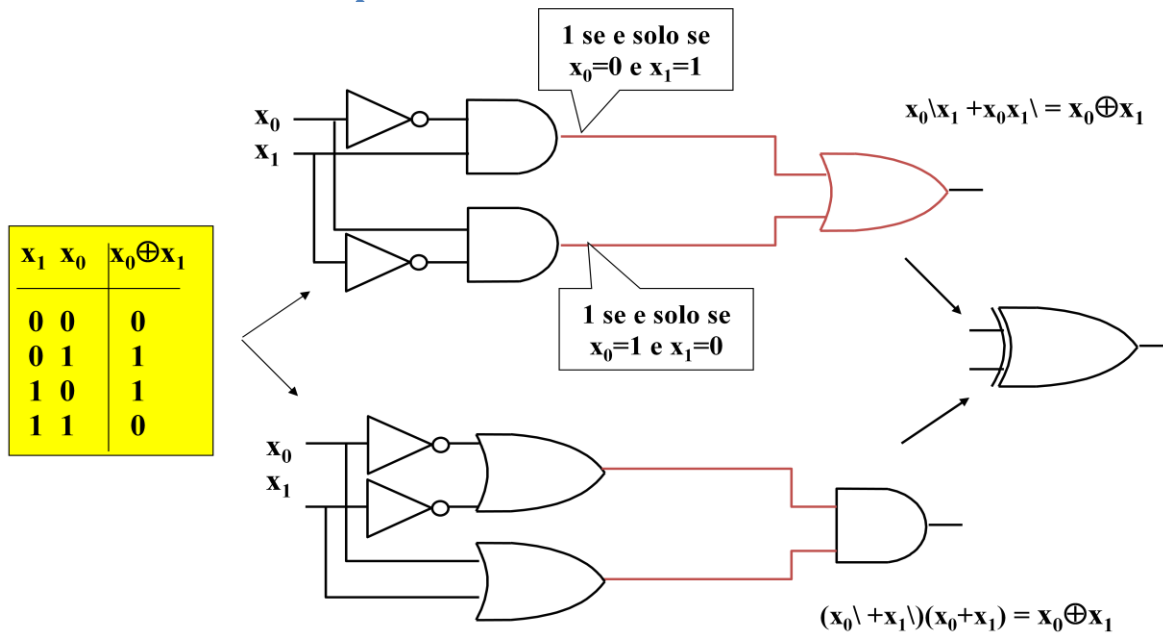
Espressione canonica SOP (Somma di Prodotti)

I^a forma canonica - Ogni funzione può essere descritta da una somma di tanti prodotti logici quante sono le configurazioni per cui vale 1. In ciascun prodotto, o mintermine, appaiono tutte le variabili, in forma o vera o complementata a seconda che nella configurazione corrispondente presentino valore 1 o valore 0.

Espressione canonica POS (Prodotto di Somme)

II^a forma canonica - Ogni funzione può essere descritta da un prodotto di tante somme logiche quante sono le configurazioni per cui vale 0. In ciascuna somma, o maxtermine, appaiono tutte le variabili, in forma o vera o complementata a seconda che nella configurazione corrispondente presentino valore 0 o valore 1.

Sintesi canonica dell'operatore EX-OR

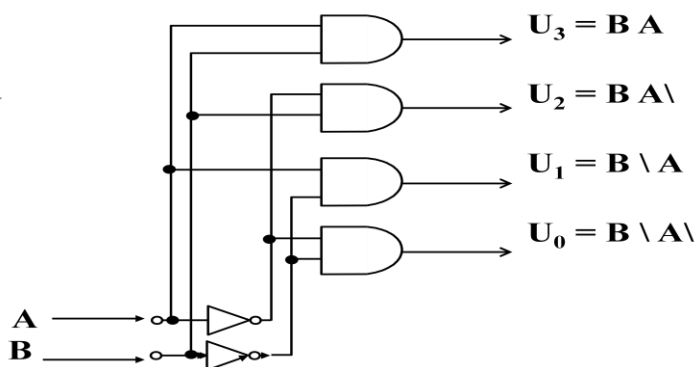


Il Decoder

Tabella della verità di un decoder a due ingressi

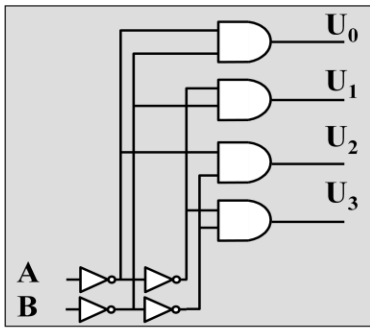
B	A	U ₀	U ₁	U ₂	U ₃
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Soluzione circuitale



Il circuito integrato MSI il DECODER

Decoder o Rete di decodifica - Rete logica combinatoria che realizza i 2^n distinti prodotti, mintermini, di n variabili ($n = 2,3,4$)



Quando EN=1, vale 1 l'uscita
il cui pedice, in decimale,
corrisponde al numero binario
in ingresso (A bit di minor peso)
Quando EN=0 le uscite sono 0

**N.B. - In questi chip le
uscite sono attive "basse"**

SN74139	U ₀
(MSI)	U ₁
EN	U ₂
A	U ₃
B	

SN74138	U ₀
(MSI)	U ₁
	U ₂
	U ₃
EN	U ₄
A	U ₅
B	U ₆
C	U ₇

SN74154	U ₀
(MSI)	U ₁
	U ₂
	U ₃
	U ₄
	U ₅
	U ₆
	U ₇
	U ₈
	U ₉
	U ₁₀
EN	U ₁₁
A	U ₁₂
B	U ₁₃
C	U ₁₄
D	U ₁₅

**Sintesi del Full Adder con Decoder e Or: con un decoder si implementano solo espressioni
SOP non ottimizzate**

$$S = AB\backslash C\backslash + A\backslash BC\backslash + A\backslash B\backslash C + ABC$$

$$R = A\backslash BC + AB\backslash C + ABC\backslash + ABC$$

