

# Laboratorio di architettura degli elaboratori

## CIRCUITI COMBINATORI

### Lezione 2

#### STRUMENTI SOFTWARE

- Logisim (<https://sourceforge.net/projects/circuit>)

#### CONTATTI

- Prof. F. Fontana ([federico.fontana@uniud.it](mailto:federico.fontana@uniud.it))
- Yuri De Pra ([yuri.depra@uniud.it](mailto:yuri.depra@uniud.it))

## Esercizio 1

Si progetti un circuito combinatorio, con tre segnali di input, che calcola la minoranza. Il circuito fornisce in uscita 1 se almeno due ingressi valgono 0, altrimenti fornisce 0 in uscita.

## Esercizio 2

Si progetti un circuito combinatorio che simula una lampadina comandata da tre diversi interruttori. I tre ingressi del circuito rappresentano lo stato degli interruttori e l'uscita rappresenta lo stato della lampadina. Il circuito deve soddisfare la condizione che ogni modifica allo stato di uno degli interruttori comporta un cambiamento dello stato della lampadina.

**Per i più volenterosi:** fornire due soluzioni per l'esercizio, la prima basata sulle sole porte AND, OR e NOT, quindi proporre una soluzione, che utilizzi meno porte, utilizzando anche le porte XOR.

## Esercizio 3

Si progetti un circuito combinatorio che riceve come ingresso due numeri binari, ALPHA e BETA, di 2 bit ciascuno (AB e CD), che generi in uscita il valore 1 se  $ALPHA \leq BETA$ , e 0 altrimenti.

Si progetti il circuito anche seguendo un approccio duale, ossia un circuito dove l'uscita è data da una porta AND che riceve come ingresso le uscite di un certo numero di porte OR.

# Correzione esercizio 1

Si progetti un circuito combinatorio, con tre segnali di input, che calcola la minoranza. Il circuito fornisce in uscita 1 se almeno due ingressi valgono 0, altrimenti fornisce 0 in uscita.

## 1. Tabella di verità

INPUT			OUTPUT
A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

2. Forma normale del circuito (blocchi di letterali in AND posti in OR):  
(sommatoria di termini, ciascuno dei quali costituito da una produttoria di letterali)

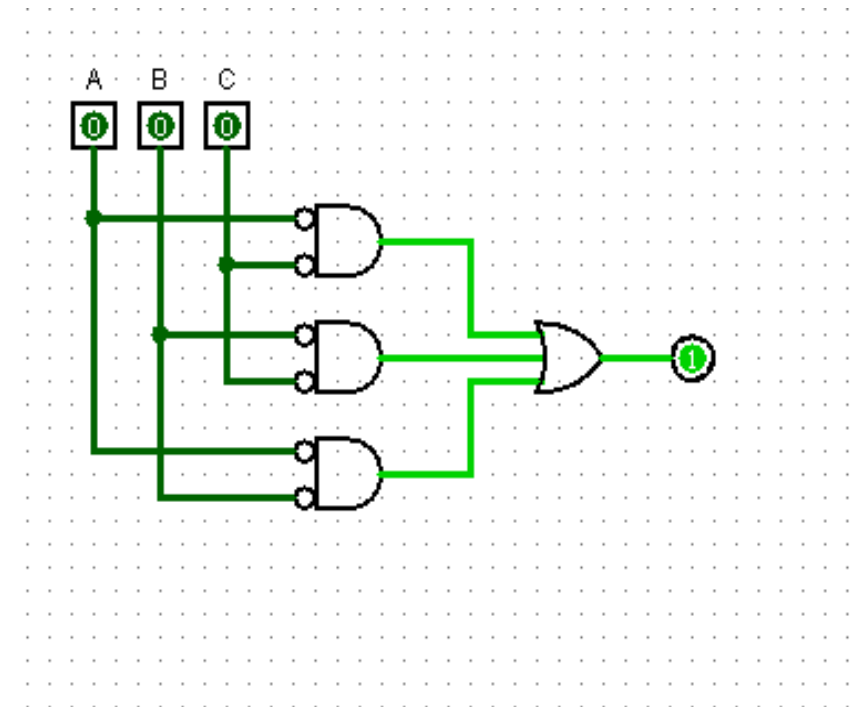
$$Y = A'B'C' + A'B'C + A'BC' + AB'C'$$

3. Mappa di Karnaugh e riduzione (quadrati potenza di 2, prendo invarianti):

$$Y = A'B' + A'C' + B'C'$$

		AB			
C		00	01	11	10
	0	1	1	0	1
	1	1	0	0	0

4. Circuito  
3 porte AND che confluiscono in una porta OR



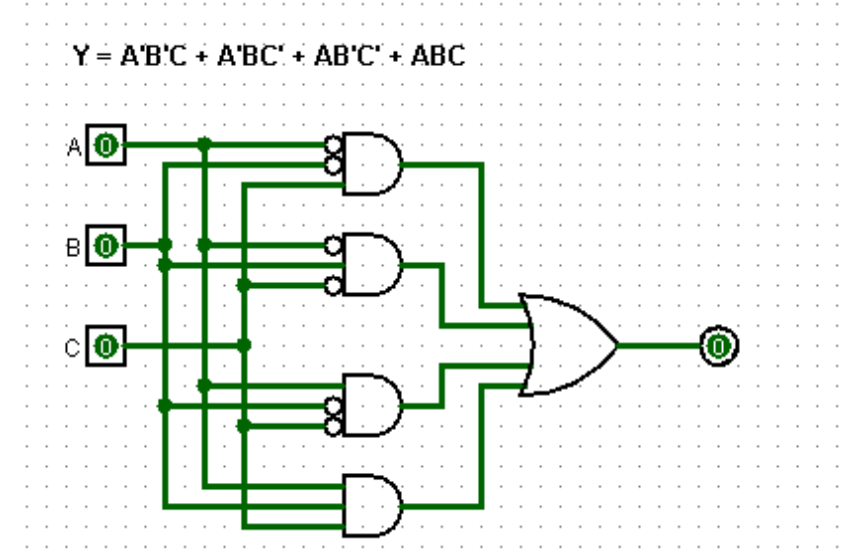
## Correzione esercizio 2

Si progetti un circuito combinatorio che simula una lampadina comandata da tre diversi interruttori. I tre ingressi del circuito rappresentano lo stato degli interruttori e l'uscita rappresenta lo stato della lampadina. Il circuito deve soddisfare la condizione che ogni modifica allo stato di uno degli interruttori comporta un cambiamento dello stato della lampadina.

**Per i più volenterosi:** fornire due soluzioni per l'esercizio, la prima basata sulle sole porte AND, OR e NOT, quindi proporre una soluzione, che utilizzi meno porte, utilizzando anche le porte XOR.

1. Tabella di verità (stato iniziale:  $A=B=C=0 \rightarrow Y=0$ )

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1



2. Forma normale  $\rightarrow Y = A'B'C + A'BC' + AB'C' + ABC$

3. Mappa di Karnaugh: non riduciamo perché quadrati da 1

		AB			
		00	01	11	10
C	0	0	1	0	1
	1	1	0	1	0

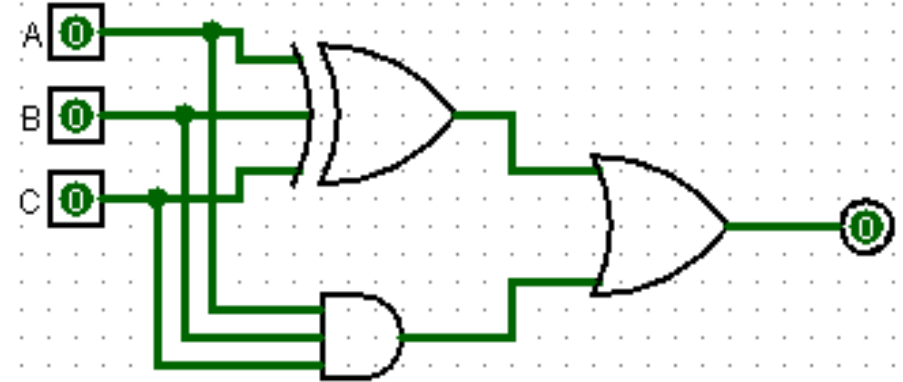
4. Forma normale della porta XOR con 3 input:

$$Y = A'B'C + A'BC' + AB'C' + ABC$$

$$Y = (\underline{A'B'C + A'BC' + AB'C'}) + ABC$$

i primi 3 termini sono assimilabili ad una XOR  
quindi sostituisco 3 porte AND con una XOR

$$\begin{aligned} Y &= A'B'C + A'BC' + AB'C' + ABC \\ &= A \text{ XOR } B \text{ XOR } C + ABC \end{aligned}$$



5. Oppure, sempre dalla forma normale:

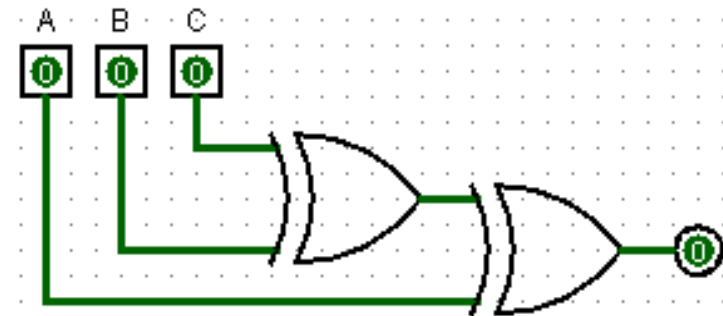
$$Y = A'B'C + A'BC' + AB'C' + ABC$$

$$= A'(\underline{B'C + BC'}) + A(\underline{B'C' + BC})$$

$$= A'(B \text{ XOR } C) + A(B \text{ XOR } C)'$$

$$= A \text{ XOR } (B \text{ XOR } C)$$

$$\begin{aligned} Y &= A'B'C + A'BC' + AB'C' + ABC \\ &= A'(B'C + BC') + A(B'C' + BC) = A \text{ XOR } (B \text{ XOR } C) \end{aligned}$$





## Correzione esercizio 3

Si progetti un circuito combinatorio che riceve come ingresso due numeri binari, ALPHA e BETA, di 2 bit ciascuno (AB e CD), che generi in uscita il valore 1 se  $ALPHA \leq BETA$ , e 0 altrimenti.

Si progetti il circuito anche seguendo un approccio duale, ossia un circuito dove l'uscita è data da una porta AND che riceve come ingresso le uscite di un certo numero di porte OR.

## 1. Mappa di Karnaugh raggruppando gli «uni»

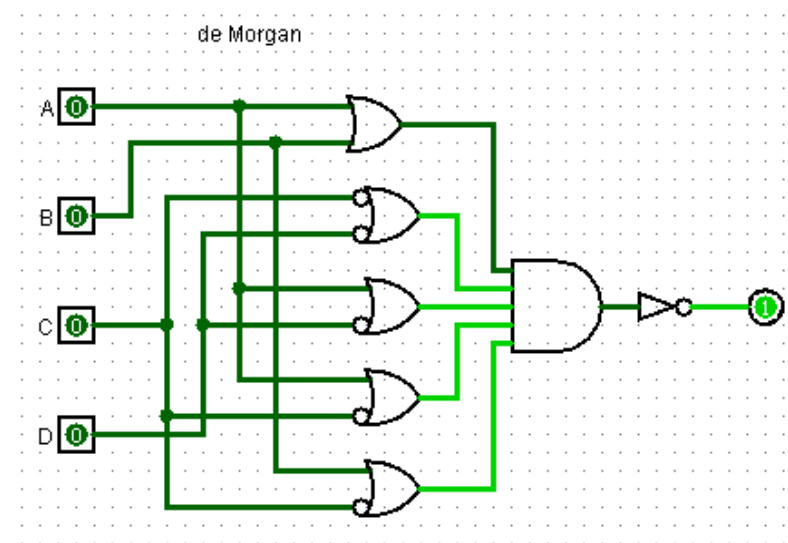
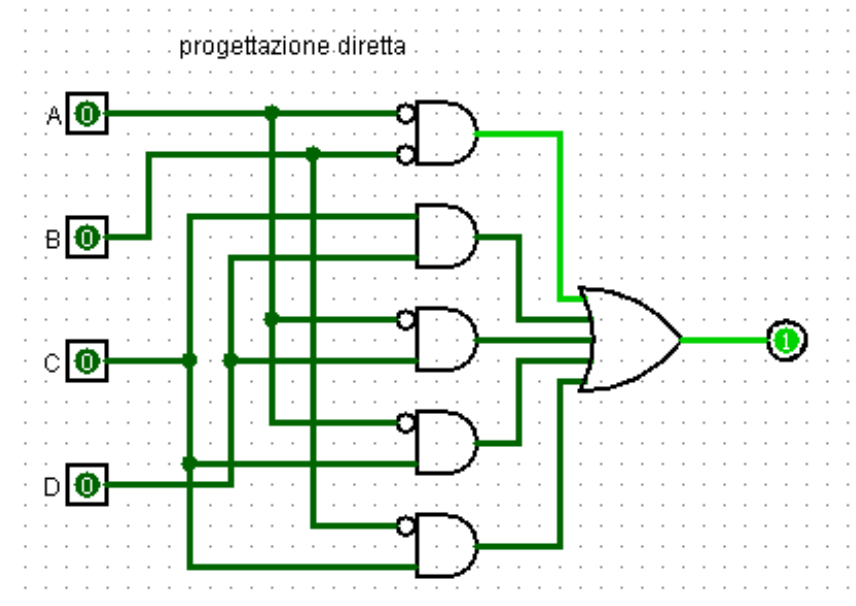
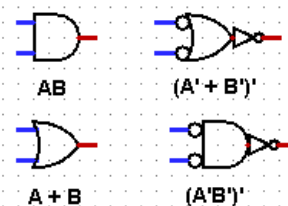
	AB (alpha)			
CD (beta)	00	01	11	10
00	1	0	0	0
01	1	1	0	0
11	1	1	1	1
10	1	1	0	1

$$Y = A'B' + CD + A'C + A'D + B'C$$

## 2. Progettazione duale 1: DeMorgan e negazione all'uscita

$$(AB)' = A' + B'; (A + B)' = A'B'$$

$$AB = (A' + B')'; A + B = (A'B')'$$



### 3. Mappa di Karnaugh raggruppando gli «zeri»

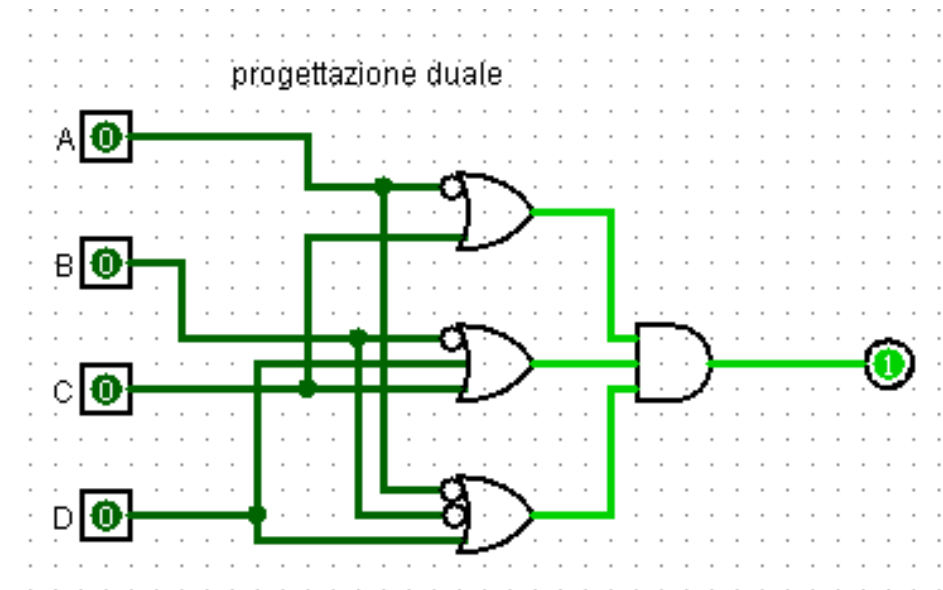
AB (alpha)					
CD (beta)	00	01	11	10	
00	1	0	0	0	
01	1	1	0	0	
11	1	1	1	1	
10	1	1	0	1	

Raggruppo gli zeri e uso espressioni duali (prodotto di somme)

$$y = a + (b'c); y' = a'(b + c')$$

$$\rightarrow Y = (A' + C) (B' + C + D) (A' + B' + D)$$

Risparmio di porte!



## Esercizio 2.1

- a) Progettare un circuito che, ricevuti 4 segnali binari (bit) in ingresso, stabilisca se questi rappresentano nella notazione binaria un numero primo (consideriamo 1 non primo). Il circuito restituisce in uscita 1 se l'input rappresenta un numero primo, mentre restituisce 0 in caso contrario.
- b) Progettare un circuito che riceva in ingresso un numero binario di 4 bit. Il circuito restituisce in uscita 1 se l'ingresso è una cifra decimale (ossia un valore tra 0 e 9) divisibile per 2 o per 5; restituisce 0 se l'ingresso è una cifra non divisibile né per 2 né per 5. Infine, nel caso in cui l'ingresso non rappresenti alcuna cifra decimale, l'uscita può assumere un valore arbitrario.

## Esercizio 2.2

- a) Costruire un multiplexer con 1 ingresso di controllo, e realizzarlo come modulo Logisim.
- b) Utilizzando tre multiplexer con 1 ingresso di controllo realizzare un multiplexer con 2 ingressi di controllo.

## Esercizio 2.3

- a) Progettare un decoder a 2 ingressi dotato di un segnale aggiuntivo di Enable. Se il segnale Enable vale 0 tutte le uscite valgono 0; se Enable vale 1 si comporta come un circuito decoder. Realizzare il circuito come modulo.
- b) Utilizzare il modulo del punto precedente per realizzare un decoder a 3 ingressi e
- c) uno a 4 ingressi.