

AE Reti Logiche (Esercitazioni)

Tommaso Zoppi
tommaso.zoppi@unifi.it



Reti Combinatorie



Esercizio 1

Si vuole progettare un circuito in grado di riconoscere i primi quattro multipli di 3 (maggiori di zero) che utilizzi il minor numero di porte logiche (AND, OR e XOR, a due ingressi, NOT ad un ingresso) Definire

- ▶ La tabella di verità di tale circuito, considerando quattro ingressi ed un'unica uscita
- ▶ La funzione SoP booleana derivante dalla tabella, eventualmente ridotta utilizzando le tecniche note di algebra booleana e ricordando il vincolo di utilizzo del minor numero possibile di porte logiche
- ▶ La rappresentazione tramite porte logiche (NOT, AND, OR, XOR) della funzione ottenuta



Esercizio 1: Tabella di Verità

N	a	b	c	d	U
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	



Esercizio 1: Tabella di Verità

N	a	b	c	d	U
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	0



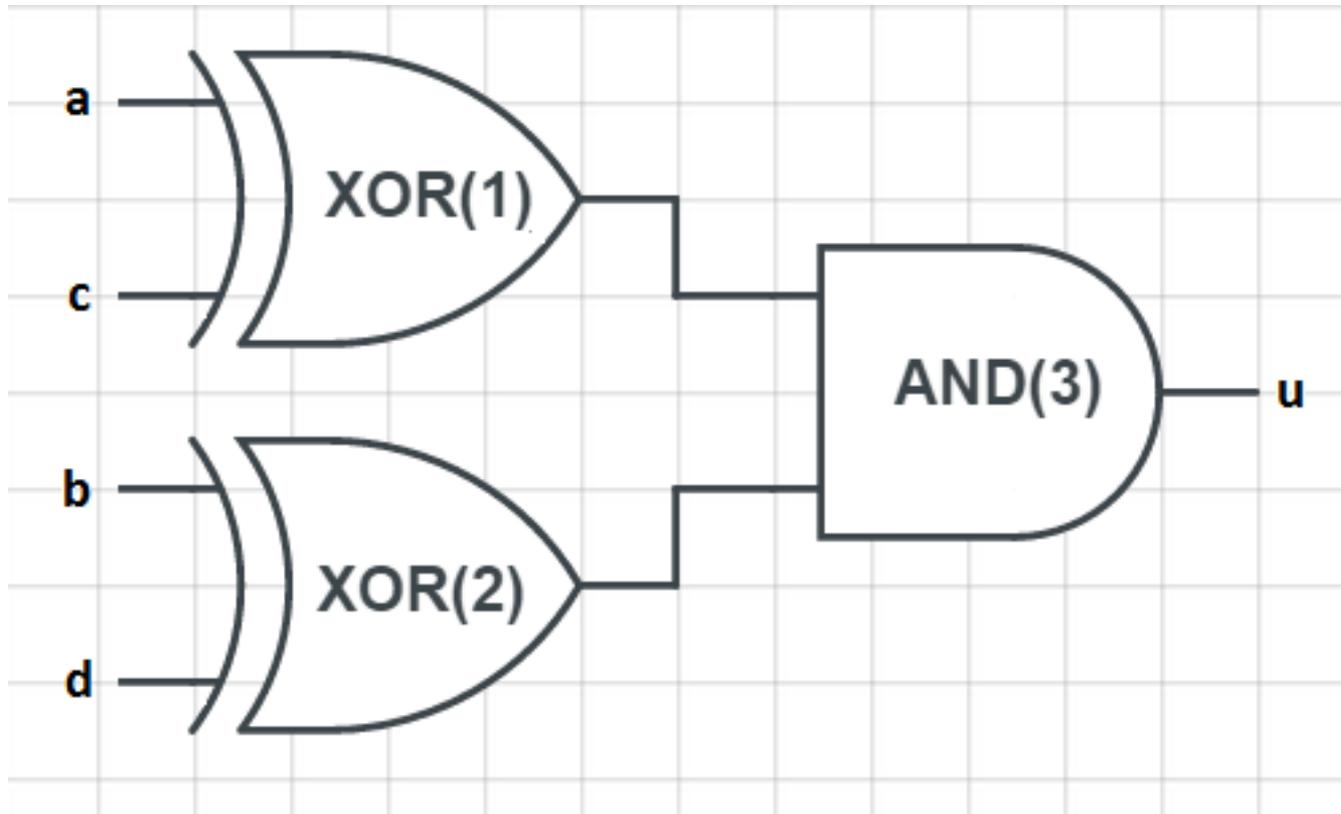
Esercizio 1: Funzione SoP booleana

► Funzione SoP booleana:

$$\begin{aligned}U &= a'b'cd + a'bcd' + ab'c'd + abc'd' = \\ &= a'c(b'd + bd') + ac'(bd' + b'd) \\ &= (ac' + a'c)(bd' + b'd) = \\ &= (a \text{ XOR } c)(b \text{ XOR } d)\end{aligned}$$



Esercizio 1: Circuito con porte logiche





Esercizio 2

Considerando una funzione logica f

$$Y = f(A, B, C, D)$$

che segnala con valore $Y=1$ tutti quei numeri che **NON** sono multipli di 3 o di 7, assumendo valore **don't care (X)** a fronte dell'input 0 ($ABCD = 0000$), definire:

- ▶ La tabella di verità della funzione f appena definita rappresentando le 4 entrate $ABCD$ ed il valore Y di output tenendo conto dei don't care (X).
- ▶ La forma SoP minima utilizzando le mappe di Karnaugh
- ▶ Disegnare il circuito corrispondente facendo uso unicamente di porte logiche AND, OR e NOT



Esercizio 2: Tabella di Verità

n	A	B	C	D	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	



Esercizio 2: Tabella di Verità

n	A	B	C	D	Y
0	0	0	0	0	X
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0



Esercizio 2: Riduzione Con mappe di Karnaugh

Y		C, D			
		00	01	11	10
A, B	00	X	1	0	1
	01	1	1	0	0
	11	0	1	0	0
	10	1	0	1	1



Esercizio 2: Riduzione Con mappe di Karnaugh

Y		C, D			
		00	01	11	10
A, B	00	X	1	0	1
	01	1	1	0	0
	11	0	1	0	0
	10	1	0	1	1

The Karnaugh map shows several groups of 1s circled in blue: a group of four 1s in the first two columns (00 and 01) for rows 00 and 01; a group of two 1s in the second column (01) for rows 00 and 01; a group of two 1s in the fourth column (10) for rows 00 and 01; a group of two 1s in the first column (00) for rows 10 and 00; a group of two 1s in the fourth column (10) for rows 10 and 00; and a group of two 1s in the fourth column (10) for rows 10 and 01.



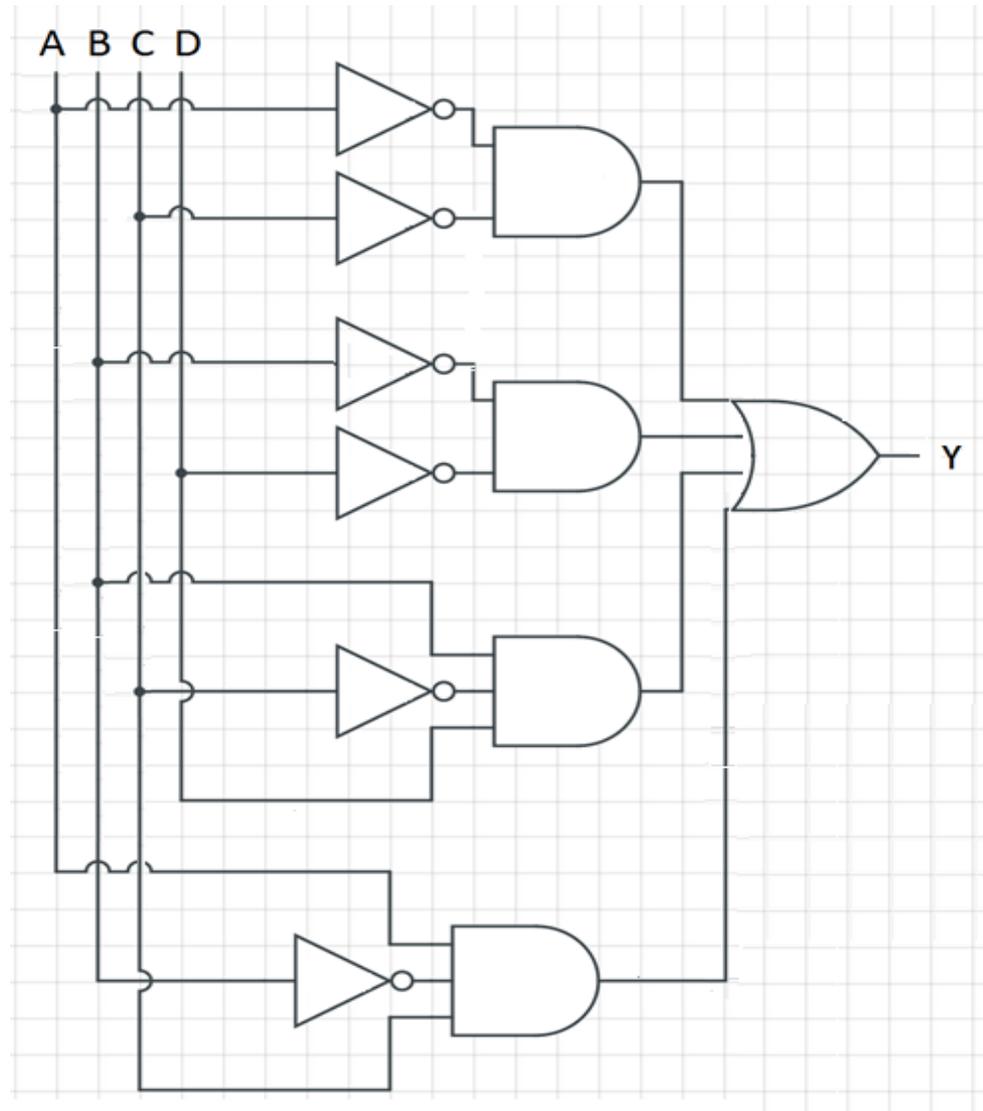
Esercizio 2: Riduzione Con mappe di Karnaugh

Y		C, D			
		00	01	11	10
A, B	00	X	1	0	1
	01	1	1	0	0
	11	0	1	0	0
	10	1	0	1	1

$$Y = f(A, B, C, D)$$
$$= B'D' + A'C' + BC'D + AB'C$$



Esercizio 2: Circuito con porte logiche





Esercizio 3

Si consideri la codifica Excess-3 (E-3), ottenuta aggiungendo $(0011)_2 = 3_{10}$ alla codifica binaria naturale di riferimento (es. $7_{10} = 0111_2 \rightarrow 1010_{E3}$).

Definire una rete logica che, a fronte di un ingresso di quattro bit ABCD corrispondenti alla codifica binaria di un numero naturale compreso tra 0 e 9, realizzi un convertitore in codifica E-3. Considerare

- ▶ come input ABCD la codifica binaria di numeri da 0 a 9 compresi. Le altre 6 combinazioni di input disponibili (corrispondenti ai decimali 10,11,12,13,14,15) non sono usate e possono essere trattate come condizioni di non specificazione, con valore **don't care**.
- ▶ quattro uscite WXYZ che rappresentano i 4 bit della codifica E-3 in output, dal più significativo al meno significativo.



Esercizio 3

- ▶ Riempire la tabella di verità del convertitore E-3
- ▶ Tramite mappe di Karnaugh, trovare una forma SoP minima per ciascuna delle uscite W, X, Y, Z.



Esercizio 3: Tabella di verità

Decimal Digit	Input BCD				Output Excess-3			
	A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0



Esercizio 3: SoP minime

- ▶ 4 uscite (W, X, Y, Z)
- ▶ 4 mappe di Karnaugh diverse
- ▶ 4 funzioni logiche risultanti



Esercizio 3: SoP minime

- ▶ 4 uscite (W, X, Y, Z)
- ▶ 4 mappe di Karnaugh diverse
- ▶ 4 funzioni logiche risultanti

4 volontari?



Esercizio 3: SoP minime delle uscite

		CD			
		00	01	11	10
AB	00				
	01		1	1	1
	11	X	X	X	X
	10	1	1	X	X

$$W = A + BC + BD$$

		CD			
		00	01	11	10
AB	00		1	1	1
	01	1			
	11	X	X	X	X
	10		1	X	X

$$X = \overline{B}C + \overline{B}D + \overline{B}\overline{C}\overline{D}$$

		CD			
		00	01	11	10
AB	00	1		1	
	01	1		1	
	11	X	X	X	X
	10	1		X	X

$$Y = CD + \overline{C}\overline{D}$$

		CD			
		00	01	11	10
AB	00	1			1
	01	1			1
	11	X	X	X	X
	10	1		X	X

$$Z = \overline{D}$$



Esercizio 3b

► Ricavare la funzione **PoS** minima per X



Esercizio 3b

- Ricavare la funzione PoS minima per X

		CD			
		00	01	11	10
AB	00	0			
	01		0	0	0
	11	X	X	X	X
	10	0		X	X

Forma PoS minima: $(B'+D')(B'+C')(B+C+D)$



Esercizio 3b

- Ricavare la funzione PoS minima per X

		CD			
		00	01	11	10
AB	00	0			
	01		0	0	0
	11	X	X	X	X
	10	0		X	X

Forma PoS minima: $(B'+D')(B'+C')(B+C+D)$

Perchè tra SoP minima e PoS minima si procede diversamente? Perchè proprio così?



Mintermini e Maxtermini

- ▶ Cosa cambia tra l'uno e l'altro?
- ▶ Siamo sicuri che sono equivalenti ai fini dell'uscita finale? Sapresti fare un esempio?



Mintermini e Maxtermini

- ▶ Cosa cambia tra l'uno e l'altro?
- ▶ Siamo sicuri che sono equivalenti ai fini dell'uscita finale? Sapresti fare un esempio?

Porta AND

Mintermini:

Maxtermini:

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1



Mintermini e Maxtermini

- ▶ Cosa cambia tra l'uno e l'altro?
- ▶ Siamo sicuri che sono equivalenti ai fini dell'uscita finale? Sapresti fare un esempio?

Porta AND

Mintermini: AB

$$\begin{aligned}\text{Maxtermini: } & (A+B)(A+B')(A'+B) = \\ & (A+AB+AB'+BB')(A'+B) = \\ & (A+AB+AB')(A'+B) = \\ & (AA'+AA'B+AA'B'+AB+ABB+ABB') = \\ & (AB+ABB) = (AB+AB) = AB\end{aligned}$$

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1



Mappe di Karnaugh: errori comuni (1)

► Ovvero: trova l'errore!

		CX			
		00	01	10	11
AB	00	X	X		
	01		1	1	
	11	X	X	X	X
	10	1	1	1	

Forma SoP minima: $AX + BX + B'C'$



Mappe di Karnaugh: errori comuni (2)

► Ovvero: trova l'errore!

		CX			
		00	01	11	10
AB	00	X	X		1
	01	1			1
	11	X	X	X	X
	10	1			1

Forma SoP minima: $X' + AB$



Mappe di Karnaugh: errori comuni (3)

► Ovvero: trova l'errore!

AB \ CX	00	01	11	10
00	X	X		
01		1	1	
11	X	X	X	X
10	1	1	1	

Forma SoP minima: $AX + BX + B'C' + AC'X'$