

Architettura degli Elaboratori I

Esercitazione 2

Fulvio D'Antonio
dantonio@di.uniroma1.it

Esercizio 1

Si semplifichino le seguenti espressioni booleane mediante le regole dell'algebra di Boole

1. $x \cdot y + \bar{x} \cdot z + \bar{y} \cdot (x + z)$
2. $\overline{(x \cdot x)}$
3. $\overline{(\bar{x} + y)} + (\bar{x} + y)$

Soluzione

- $$xy + \bar{x}z + \bar{y}x + \bar{y}z =$$
1. $x + z(\bar{x} + \bar{y}) =$
 $(x + z)(x + \bar{x} + \bar{y}) =$
 $x + z$

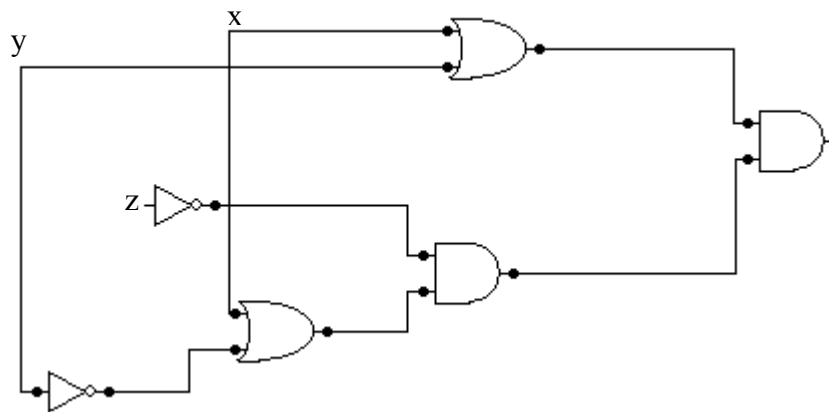
Qui fra gli altri teoremi, è stato applicato quello della proprietà distributiva dell'OR rispetto all'AND (vedere lucidi e appunti)

2. $\overline{xx} = \bar{0} = 1$

3. $\overline{(\bar{x} + y)} + (\bar{x} + y) = \bar{a} + a = 1$

Esercizio 2

Sia dato il seguente schema circuitale:



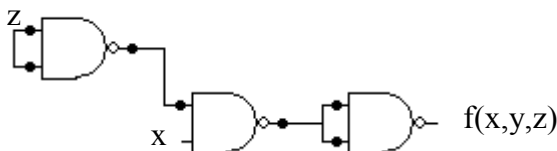
Ottenere da tale schema una espressione booleana (EB) corrispondente e la relativa tabella di verità. Realizzare inoltre lo stesso schema facendo uso di sole porte NAND.

Soluzione:

etichettando gli archi con le espressioni booleane che rappresentano i segnali via via elaborati dalle varie porte logiche, si ottiene per il segnale di uscita:

$$f(x, y, z) = (x + y)(\bar{z}(x + \bar{y})) = \bar{z}((x + y)(x + \bar{y})) = \bar{z}x$$

Quindi basta simulare una porta NOT ed una porta AND tramite porte NAND per ottenere:



Esercizio 3

Dati due valori interi $A = a_1a_0$ e $B = b_1b_0$ rappresentati con 2 bit in codifica binaria si vuole calcolare il numero $Y = y_3y_2y_1y_0$ rappresentato con 4 bit nella codifica in complemento a due.

$$Y = F(A,B) = A * B - 2 * A + 2 * B$$

Nota: nei casi in cui la funzione produca un valore al di fuori del range di rappresentazione si consideri tale valore “non definito” (don’t care)

Si realizzi il circuito combinatorio che realizza F mediante:

- ROM
- PLA
- Multiplexers 8-a-1

Soluzione:

Questa tabella di verità, per chiarezza, mostra gli interi che costituiscono gli addendi di Y ed Y anche in codice decimale.

b ₁ b ₀ a ₁ a ₀	y ₃ y ₂ y ₁ y ₀	AxB	-2A	+2B	Y
0000	0000	0	0	0	0
0001	1110	0	-2	0	-2
0010	1100	0	-4	0	-4
0011	1010	0	-6	0	-6
0100	0010	0	0	2	2
0101	0001	1	-2	2	1
0110	0000	2	-4	2	0
0111	0001	3	-6	2	-1
1000	0100	0	0	4	4
1001	0100	2	-2	4	4
1010	0100	4	-4	4	4
1011	0100	6	-6	4	4
1100	0110	0	0	6	6
1101	0111	3	-2	6	7
1110	xxxx	6	-4	6	8
1111	xxxx	9	-6	6	9

Notate che, poichè 8 e 9 non sono rappresentabili con 4 bit in complemento a due, la funzione di uscita binaria in corrispondenza di quei valori è indefinita, come richiesto dal testo del problema.

Le mappe di karnaugh di y₀ e y₁ sono:

b ₁ b ₀ \a ₁ a ₀	00	01	11	10	00	01	11	10
00	0	0	0	0	0	1	1	0
01	0	1	1	0	1	0	0	0
11	0	0	0	0	0	0	0	0
10	0	1	X	x	1	1	x	x

$$y_0 = \overline{b_1}b_0a_0 + b_1\overline{b_0}a_0$$

$$y_1 = \overline{b_1}b_0\overline{a_1}a_0 + \overline{b_0}a_0$$

b ₁ b ₀ \a ₁ a ₀	00	01	11	10	00	01	11	10
00	0	1	0	1	0	1	0	1
01	0	0	0	0	0	0	0	0
11	1	1	X	x	0	0	x	x
10	1	1	1	1	0	0	0	1

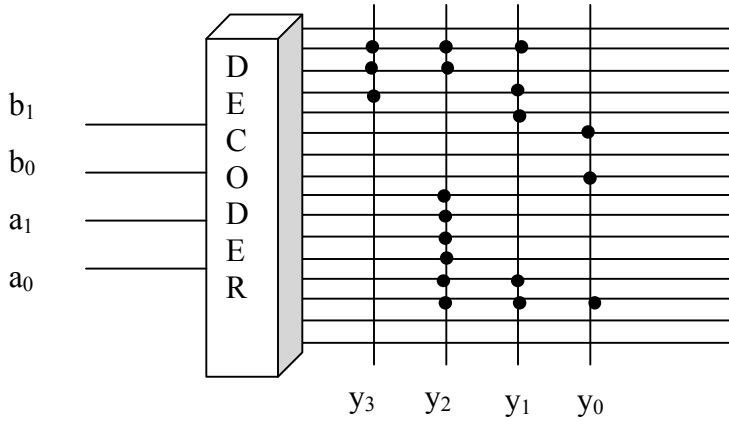
Dalle mappe di Karnaugh di y₂ e y₃ si ottiene:

$$y_2 = b_1 + \overline{b_0 a_1 a_0} + \overline{b_0 a_1 a_0}$$

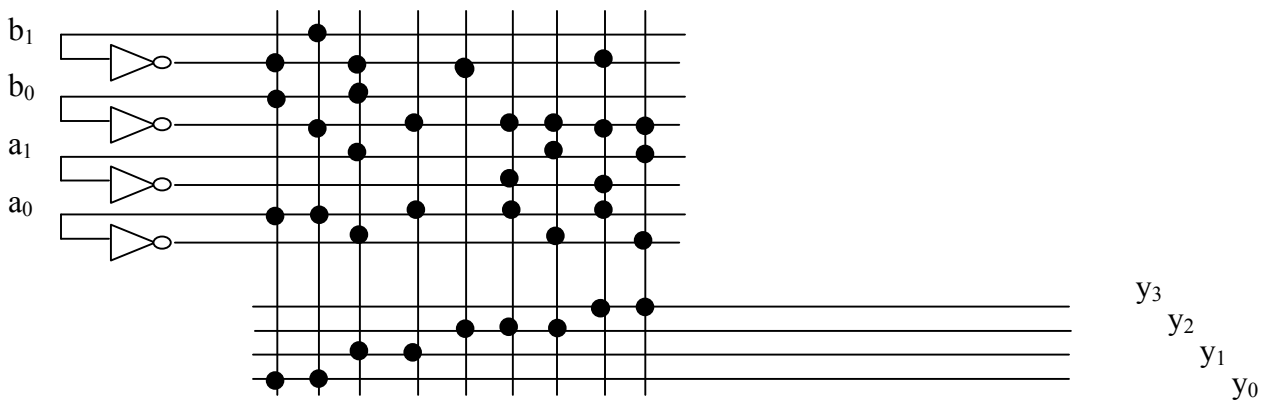
$$y_3 = \overline{b_1 b_0 a_1 a_0} + \overline{a_1 a_0 b_0}$$

Realizzazione dei circuiti:

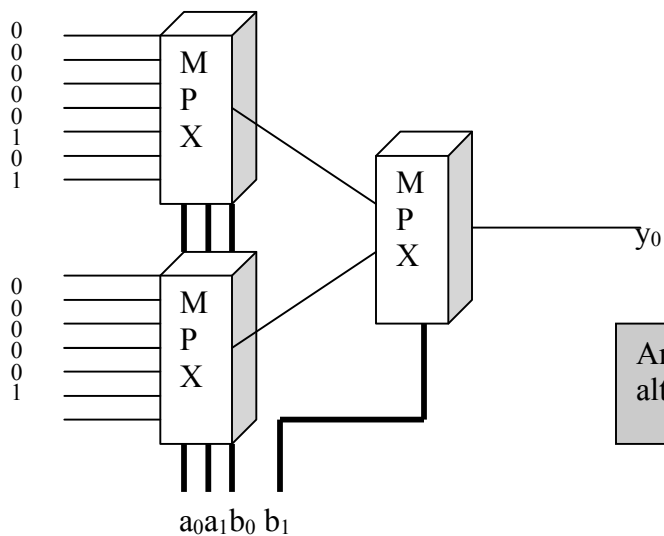
Mediante ROM



Mediante PLA



Mediante Multiplexer



Analogamente si disegnano i MPX per le altre funzioni y_i