SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Esercizi di sintesi di reti combinatorie

Prof. Daniele Gorla

Specifica

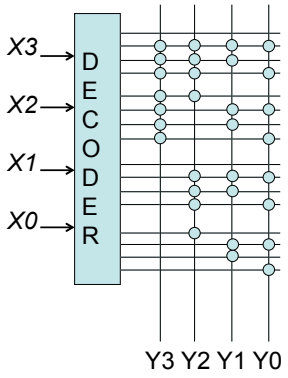
Si realizzi in tutti i modi possibili (porte logiche, ROM, PLA e MUX di tutte le dimensioni possibili) un circuito che calcoli l'opposto di un numero intero da 4 bit rappresentato in Ca2.

Sequenza non utilizzata nel Ca2 perché non ha l'opposto rappresentabile nel formato specificato

$x_3\,x_2\,x_1\,x_0$	$y_3\,y_2\,y_1\,y_0$
0 0 0 0	0 0 0 0
0 0 0 1	1 1 1 1
0 0 1 0	1 1 1 0
0 0 1 1	1 1 0 1
0 1 0 0	1 1 0 0
0 1 0 1	1 0 1 1
0 1 1 0	1 0 1 0
0 1 1 1	1 0 0 1
1 0 0 0	- - - -
1 0 0 1	0 1 1 1
1 0 1 0	0 1 1 0
1 0 1 1	0 1 0 1
1 1 0 0	0 1 0 0
1 1 0 1	0 0 1 1
1 1 1 0	0 0 1 0
1 1 1 1	0 0 0 1

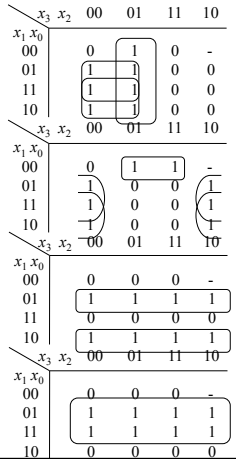
Realizzazione tramite ROM

$x_3\,x_2\,x_1\,x_0$	$y_3\,y_2\,y_1\,y_0$
0 0 0 0	0 0 0 0
0 0 0 1	1 1 1 1
0 0 1 0	1 1 1 0
0 0 1 1	1 1 0 1
0 1 0 0	1 1 0 0
0 1 0 1	1 0 1 1
0 1 1 0	1 0 1 0
0 1 1 1	1 0 0 1
1 0 0 0	- - - -
1 0 0 1	0 1 1 1
1 0 1 0	0 1 1 0
1 0 1 1	0 1 0 1
1 1 0 0	0 1 0 0
1 1 0 1	0 0 1 1
1 1 1 0	0 0 1 0
1 1 1 1	0 0 0 1



Porte logiche

$x_2\,x_1\,x_0$	$y_3\,y_2\,y_1\,y_0$
0 0 0	0 0 0 0
0 0 1	1 1 1 1
0 1 0	1 1 1 0
0 1 1	1 1 0 1
1 0 0	1 1 0 0
1 0 1	1 0 1 1
1 1 0	1 0 1 0
1 1 1	1 0 0 1
0 0 0	- - - -
0 0 1	0 1 1 1
0 1 0	0 1 1 0
0 1 1	0 1 0 1
1 0 0	0 1 0 0
1 0 1	0 0 1 1
1 1 0	0 0 1 0
1 1 1	0 0 0 1



y3 = x3x2+x3x1+x3x0
= x3(x2+x1+x0)

y2 = x2x0+x2x1+x2x1x0
= x2(x0+x1)+x2x1x0
= x2 XOR (x1+x0)

y1 = x1x0+x1x0
= x1 XOR x0

y0 = x0

PLA

SAPIENZA Università di Roma Dipartimento di Informatica

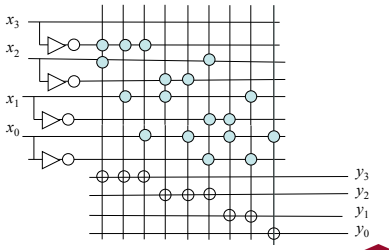
$x_3\,x_2\,x_1\,x_0$	$y_3\,y_2\,y_1\,y_0$
0 0 0 0	0 0 0 0
0 0 0 1	1 1 1 1
0 0 1 0	1 1 1 0
0 0 1 1	1 1 0 1
0 1 0 0	1 1 0 0
0 1 0 1	1 0 1 1
0 1 1 0	1 0 1 0
0 1 1 1	1 0 0 1
1 0 0 0	- - - -
1 0 0 1	0 1 1 1
1 0 1 0	0 1 1 0
1 0 1 1	0 1 0 1
1 1 0 0	0 1 0 0
1 1 0 1	0 0 1 1
1 1 1 0	0 0 1 0
1 1 1 1	0 0 0 1

y3 = x3x2+x3x1+x3x0

y2 = x2x0+x2x1+x2x1x0

y1 = x1x0+x1x0

y0 = x0

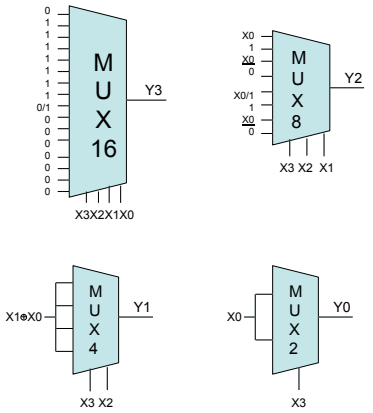


5

MUX

SAPIENZA Università di Roma Dipartimento di Informatica

$x_3\,x_2\,x_1\,x_0$	$y_3\,y_2\,y_1\,y_0$
0 0 0 0	0 0 0 0
0 0 0 1	1 1 1 1
0 0 1 0	1 1 1 0
0 0 1 1	1 1 0 1
0 1 0 0	1 1 0 0
0 1 0 1	1 0 1 1
0 1 1 0	1 0 1 0
0 1 1 1	1 0 0 1
1 0 0 0	- - - -
1 0 0 1	0 1 1 1
1 0 1 0	0 1 1 0
1 0 1 1	0 1 0 1
1 1 0 0	0 1 0 0
1 1 0 1	0 0 1 1
1 1 1 0	0 0 1 0
1 1 1 1	0 0 0 1



6

Confronto

SAPIENZA Università di Roma Dipartimento di Informatica

ROM: - DECOD 4-a-16: 4 NOT e 16 AND
- 1 diodo per ogni "1" → 7+8+8+8 = 31
TOTALE: 51 porte/diodi

Porte: 1 NOT + 1 AND + 2 OR (una si riusa) + 2 XOR = 6 porte

PLA: 4 NOT + 10 AND + 6 OR = 20 porte

MUX: - MUX 16-a-1 = 20 (DEC 4-a-16) + 16 AND + 15 OR
- MUX 8-a-1 = 11 (DEC 3-a-8) + 8 AND + 7 OR
+ 1 NOT per realizzare la funzione
- MUX 4-a-1 = 6 (DEC 2-a-4) + 4 AND + 3 OR
+ 1 XOR per la funzione
- MUX 2-a-1 = 3 (DEC 1-a-2) + 2 AND + 1 OR
TOTALE = 51+27+14+6 = 98 porte

7