

Compito A

Es. 1: Esprimi in complemento a due il numero decimale - 61,81 arrestandosi al 6° bit dopo la virgola.

Esprimi lo stesso numero normalizzato in virgola mobile. Quanti bit sono necessari complessivamente nel primo e nel secondo caso, assegnando 7 bit alla mantissa (espressa sempre in Ca2)?

Es. 2: Progetta un circuito combinatorio che riceve in ingresso 7 bit $X_6X_5\dots X_0$ e produce in uscita 8 bit di cui i 7 più significativi $Y_7Y_6\dots Y_1$ sono uguali a $X_6X_5\dots X_0$, rispettivamente, e $Y_0=1$ se $X_6X_5\dots X_0$ contiene un numero di "1" pari, mentre $Y_0=0$ altrimenti.

Es. 3: Disegna a livello di porte logiche l'architettura interna di un Multiplexer con 8 bit di ingresso.

Soluzioni

Esercizio 1

0111101,110011 (+61,81) CA2: $1000010,001100+1=1000011,001100$ (-61,81)

Per passare alla forma in virgola mobile, bisogna dapprima normalizzare, il che si ottiene moltiplicando per 2^6 (il bit più significativo è il segno):

$\langle 1 \rangle \langle 000011,001100 \rangle \langle 0 \rangle$ (questa è la forma non normalizzata: $1000011,001100 \times 2^0$)

per normalizzare si deve spostare la virgola di sei posizioni a sinistra ottenendo: $1,000011001100 \times 2^6$ (il bit prima della virgola è il segno, e la mantissa è in Ca2).

Poiché voglio assegnare solo 7 bit alla mantissa, ed ho bisogno di 4 bit (in Ca2) per esprimere 6, ottengo:

$\langle 1 \rangle \langle 0000110 \rangle \langle 0110 \rangle$ dunque servono 12 bit.

Notate che il procedimento che parte dal numero decimale: $0,6182 \times 10^2$ e che converte separatamente 0,6182 e +2 porterebbe ad un numero la cui mantissa e esponenete sono binari, ma la cui base è decimale!!

Esercizio 2

Evidentemente, $Y_i = X_{i-1}$ $i > 0$

$$Y_0 = \begin{matrix} \bar{X}_3 \bar{X}_2 \bar{X}_1 \bar{X}_0 & \bar{X}_3 \bar{X}_2 X_1 X_0 & \bar{X}_3 X_2 \bar{X}_1 X_0 & \bar{X}_3 X_2 X_1 \bar{X}_0 & X_3 X_2 \bar{X}_1 \bar{X}_0 \\ X_3 X_2 X_1 X_0 & X_3 \bar{X}_2 \bar{X}_1 X_0 & X_3 \bar{X}_2 X_1 \bar{X}_0 & & \end{matrix}$$

1		1	
	1		1
1		1	
	1		1

Nota: vi ricordate la definizione di parità?? Zero è pari!!

La EB di Y_0 può essere espressa in modo più compatto usando porte XOR (note: NON semplificata, poiché una porta XOR è un integrato realizzato con invertitori, porte AND e porte OR):

$$Y_0 = X_3 \text{ XOR } X_2 \text{ XOR } X_1 \text{ XOR } X_0$$

E di conseguenza lo schema circuitale è molto semplice.

Esercizio 3:

Scrivo solo le espressioni booleane per n ingressi ($I_0, I_1 \dots I_{n-1}$), una uscita Y, e $\log_2 n$ segnali di controllo $C_0, C_1 \dots C_{\log_2 n}$:

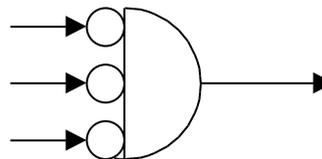
$$Y = X_0 \text{ OR } X_2 \dots X_{n-1} \text{ (porta OR)}$$

$X_i = I_i \text{ Di}$ ($i=0..n-1$) (n porte AND ciascuna delle quali ha in ingresso un input e la corrispondente uscita del decodificatore)

$$D_i = B_{\log_2 n} \dots B_0$$

Dove $B_{\log_2 n} \dots B_0$ è il mintermine corrispondente alla i-esima riga della tabella di verità del decoder. (ovvero, $B_j = D_j$ se nella i-esima riga $D_i = 1$, ed è $\text{not}(D_j)$ se $D_j = 0$).

Per $n=8$ e $\log_2 n=3$, ad esempio, $D_0 = \bar{C}_2 \bar{C}_1 \bar{C}_0$



Compito B

Es 1:

- Semplifica l'espressione: $\bar{X}\bar{Y}\bar{Z} + \bar{X}Y\bar{Z} + X\bar{Y}\bar{Z} + XYZ$
- Converti in binario ed esegui $(3, \frac{1}{4}) + (5, \frac{3}{4})$

Es 2: Progetta un sommatore parallelo di interi a 4 bit, utilizzando solo porte NAND

Es 3: Progetta un circuito che riceve 3 ingressi $X_2X_1X_0$ e produce $Y=1$ se $X_0=X_1 \ X_2$ (OR) oppure se $X_1=X_0 \ X_2$ (AND)

Soluzioni:

Es. 1

$$F = \bar{Z}$$

$$3,1/4 = 3,25_{10} = 011,01_2$$

$$5,3/4 = 5,75_{10} = 101,11_2$$

$$011,01 + 101,11 = 1001,00$$

le conversioni si effettuano col metodo iterativo spiegato a lezione e nell' esercitazione 1 (in rete).

Es. 2

Per lo schema del sommatore realizzato con full adder, si veda sugli appunti in rete.

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

$$C_i = A_i C_{i-1} + B_i C_{i-1} + A_i B_i$$

Con porte NAND si ottiene, applicando ripetutamente De Morgan e l'assioma della doppia complementazione:

$$S_i = \overline{\overline{(C \bar{A} \bar{B})} \overline{(\bar{C} \bar{A} \bar{B})} \overline{(C \bar{A} \bar{B})} \overline{(\bar{C} \bar{A} \bar{B})}}$$

dove per brevità sono omissi i pedici

$$C_i = \overline{(\bar{A} \bar{C})} \overline{(\bar{B} \bar{C})} \overline{(\bar{A} \bar{B})}$$

Alternativamente, basta rimpiazzare ogni AND con due NAND in sequenza, ogni OR con un NAND i cui ingressi corrispondono agli ingressi del OR negati. La negazione si ottiene con un NAND i cui ingressi sono congiunti.. (Si vedano anche qui gli appunti e il Tocci).

Esercizio 3

X2X1X0	Condizione 1: X0=X1 X2 (1=TRUE, 0=FALSE)	Condizione 2: X1=X0 X2 (1=TRUE, 0=FALSE)	Y= Condizione 1 Condizione 2
000	1	1	1
001	0	1	1
010	0	0	0
011	1	0	1
100	0	1	1
101	1	0	1
110	0	0	0
111	1	1	1

$Y=X0+\text{not}(X1)$

Compito C

Es. 1:

- a) Esprimi in complemento a due il numero decimale - 47,78125 avendo a disposizione 8 bit per la parte intera e 4 bit per la parte frazionaria.
- b) Cambierebbe il risultato avendo a disposizione 6 bit per la parte frazionaria?
- c) Avendo invece 8 bit, si otterrebbe un risultato diverso rispetto alla soluzione del punto precedente? Perché?

Es. 2:

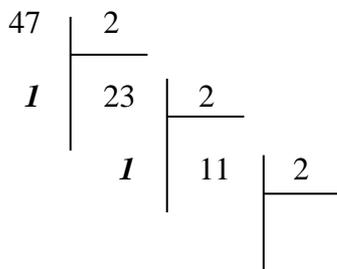
Si progetti un circuito che, presi in input due naturali A e B rappresentati con 2 bit, dà in output A+B, se tale numero è rappresentabile con 2 bit, altrimenti dà i due bit più significativi di A+B. (Si risolva l'esercizio dando prima la tavola di verità della funzione desiderata, la si minimizzi con la tecnica di Karnaugh e in seguito con gli assiomi dell'algebra di Boole per ottenere la minima espressione equivalente, ed infine se ne disegni il circuito ottenuto).

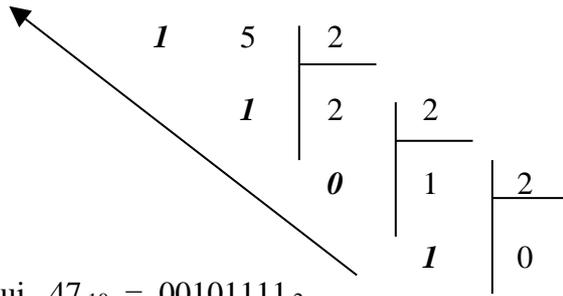
Es. 3:

- a) Si esponga il funzionamento logico di un PLA.
- b) E' corretto dire che le prestazioni di un PLA in termini di tempo di risposta sono buone quanto quelle del miglior circuito combinatorio progettato per la stessa espressione booleana? Perché?

C.1

- a) Anzitutto convertito in binario 47:





da cui $47_{10} = 00101111_2$

Converto in binario 0,78125:

$$0,78125 * 2 = 1,5625$$

$$0,5625 * 2 = 1,125$$

$$0,125 * 2 = 0,25$$

$$0,25 * 2 = 0,5$$

da cui $0,78125_{10} = 0,1100_2$

Passo alla realizzazione del complemento a due:

00101111,1100

11010000,0011

11010001,0011

per cui $-47,78125_{10} = 11010001,0011_2$

c) Converto in binario 0,78125, proseguendo oltre il 4° bit frazionario:

$$0,78125 * 2 = 1,5625$$

$$0,5625 * 2 = 1,125$$

$$0,125 * 2 = 0,25$$

$$0,25 * 2 = 0,5$$

$$0,5 * 2 = 1,0$$

L' algoritmo termina perché la parte frazionaria del risultato è 0

da cui $0,78125_{10} = 0,110010_2$

Quindi si ottiene $-47,78125_{10} = 11010001,001101_2$, diverso dal risultato del punto a)

d) Per tutte le rappresentazioni che assegnano più di 5 bit alla parte frazionaria, il risultato è sempre lo stesso poiché non si fa altro che aggiungere zeri non significativi dal sesto bit frazionario in poi (prima del passaggio al complemento a due).

C.2

a_1	a_0	b_1	b_0	c_1	c_0
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	1	1
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	1	1
0	1	1	1	1	0
1	0	0	0	1	0

$$\begin{array}{cccc|cc}
 1 & 0 & 0 & 1 & 1 & 1 \\
 1 & 0 & 1 & 0 & 1 & 0 \\
 1 & 0 & 1 & 1 & 1 & 0 \\
 1 & 1 & 0 & 0 & 1 & 1 \\
 1 & 1 & 0 & 1 & 1 & 0 \\
 1 & 1 & 1 & 0 & 1 & 0 \\
 1 & 1 & 1 & 1 & 1 & 1
 \end{array}$$

La mappa di K. per c_l è :

$b_1 b_0$	00	01	11	10
$a_1 a_0$				
00	0	0	1	1
01	0	1	1	1
11	1	1	1	1
10	1	1	1	1

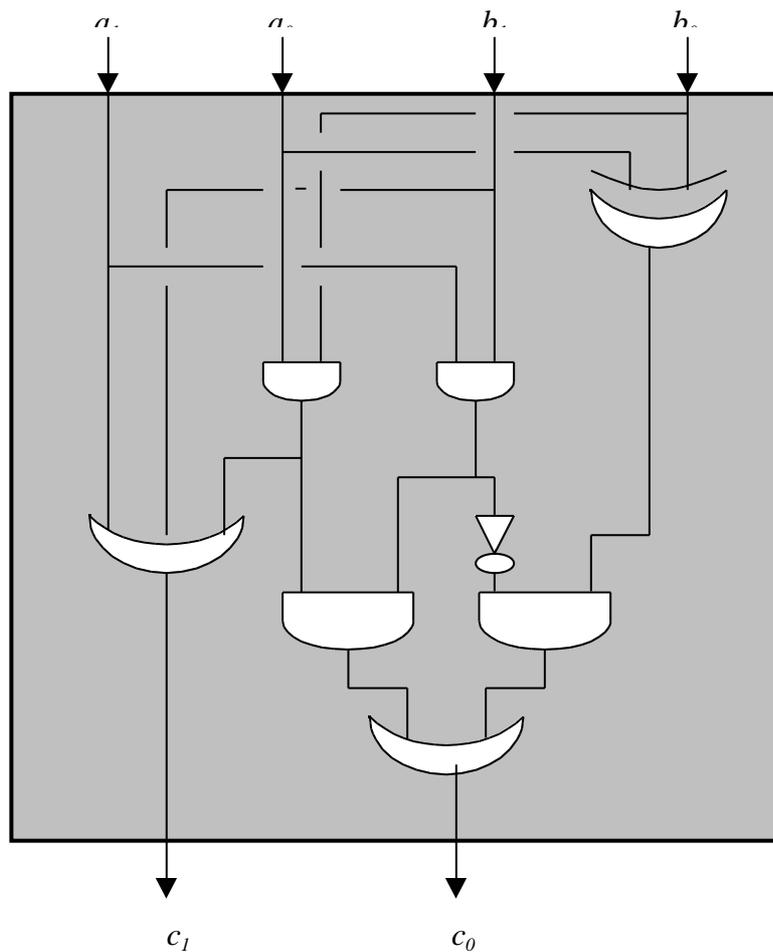
da cui $C_{min}^1 = a_1 + b_1 + a_0 b_0$, che non è ulteriormente semplificabile.

La mappa di K. per c_0 è :

$b_1 b_0$	00	01	11	10
$a_1 a_0$				
00	0	1	1	0
01	1	0	0	1
11	1	0	1	0
10	0	1	0	0

$$\begin{aligned}
 \text{da cui } C_{min}^2 &= \bar{a}_1 \bar{a}_0 b_0 + \bar{a}_0 \bar{b}_1 b_0 + \bar{a}_1 a_0 \bar{b}_0 + a_0 \bar{b}_1 \bar{b}_0 + a_1 a_0 b_1 b_0 \\
 &= \bar{a}_1 (\bar{a}_0 b_0 + a_0 \bar{b}_0) + \bar{b}_1 (\bar{a}_0 b_0 + a_0 \bar{b}_0) + a_1 a_0 b_1 b_0 \\
 &= (\bar{a}_1 + \bar{b}_1) (a_0 \text{ XOR } b_0) + a_1 a_0 b_1 b_0 = \overline{(a_1 \cdot b_1)} (a_0 \text{ XOR } b_0) + a_1 a_0 b_1 b_0
 \end{aligned}$$

Pertanto il circuito desiderato è:



C.3

E' corretto rispondere sia sì che no. Infatti il PLA realizza la miglior EB in forma SOP; tale espressione può essere minimizzata ulteriormente usando porte che non siano solo AND, OR e NOT, cioè usando porte NAND, NOR e XOR. In realtà, però, queste ultime sono ottenute come composizione delle prime, e pertanto il numero di porte non cambia; cambia il fatto che, essendo le porte composte circuiteria integrata, si risparmia sulle connessioni tra le porte e quindi il tempo di attraversamento dei circuiti con porte composte (XOR, ad esempio) risulta inferiore del corrispondente PLA.

Compito D

Es 1:

e) Semplifica l'espressione: $x y z + x \bar{y} z + \bar{x} y z + \bar{x} \bar{y} z$

f) Converti in binario ed esegui $2,25 + 3,75$ usando 2 bit per la parte intera e due bit per la parte frazionaria.

Es 2:

Si descriva a parole l'algoritmo per ottenere la forma canonica disgiuntiva di una funzione

$$f : \{0,1\}^n \rightarrow \{0,1\}$$

Es 3:

Si dia un circuito combinatorio che, preso un numero intero A di 4 bit rappresentato in complemento a due, restituisca l'intero B tale che:

$$B = \begin{cases} A & \text{se A è pari} \\ A-1 & \text{se A è dispari ed è positivo} \\ A+1 & \text{se A è dispari ed è negativo} \end{cases}$$

Si assuma che A sia sempre diverso dalla sequenza 1000. (Si risolva l'esercizio dando prima la tavola di verità della funzione desiderata, la si minimizzi con la tecnica di Karnaugh e in seguito con gli assiomi dell'algebra di Boole per ottenere la minima espressione equivalente, ed infine se ne disegni il circuito ottenuto).

D.1

a) $x y z + x \bar{y} z + \bar{x} y z + \bar{x} \bar{y} z = x z + \bar{x} z = z$

b) $2_{10} = 10_2$, $3_{10} = 11_2$, $0,25_{10} = 0,01_2$, $0,75_{10} = 0,11_2$

Quindi
$$\begin{array}{r} 10,01 \\ + \\ 11,11 \\ \hline \end{array}$$

$$110,00$$

Con il numero di bit disponibili il risultato è 10,00 errato a causa di un overflow.

D.3

a_3	a_2	a_1	a_0	b_3	b_2	b_1	b_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	0
0	1	0	0	0	1	0	0
0	1	0	1	0	1	0	0
0	1	1	0	0	1	1	0
0	1	1	1	0	1	1	0
1	0	0	0	-	-	-	-
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	0
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	0
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	0
1	1	1	1	0	0	0	0

La mappa di K. per b_3 è :

		$a_1 a_0$			
		00	01	11	10
$a_3 a_2$	00	0	0	0	0
	01	0	0	0	0
	11	1	1	0	1
	10	-	1	1	1

da cui $B^3_{min} = a_3 \bar{a}_2 + a_3 \bar{a}_1 + a_3 \bar{a}_0 = a_3 (\bar{a}_2 + \bar{a}_1 + \bar{a}_0) = a_3 \cdot \overline{(a_2 \cdot a_1 \cdot a_0)}$

La mappa di K. per b_2 è :

		$a_1 a_0$			
		00	01	11	10
$a_3 a_2$	00	0	0	0	0
	01	1	1	1	1
	11	1	1	0	1
	10	-	0	1	0

da cui $B_{min}^2 = \bar{a}_2 \bar{a}_3 + \bar{a}_2 \bar{a}_1 + \bar{a}_2 \bar{a}_0 + \bar{a}_3 \bar{a}_2 a_1 a_0 = \bar{a}_2 (\bar{a}_3 + \bar{a}_1 + \bar{a}_0) + \bar{a}_3 \bar{a}_2 a_1 a_0$
 $= \bar{a}_2 \text{ XOR } (a_3 \cdot a_1 \cdot a_0)$

La mappa di K. per b_1 è :

		$a_1 \ a_0$			
		00	01	11	10
$a_3 \ a_2$					
	00	0	0	1	1
	01	0	0	1	1
	11	0	1	0	1
	10	-	1	0	1

da cui $B_{min}^1 = \bar{a}_3 a_1 + a_1 \bar{a}_0 + \bar{a}_3 \bar{a}_1 a_0 = a_1 (\bar{a}_3 + \bar{a}_0) + \bar{a}_3 \bar{a}_1 a_0 = a_1 \text{ XOR } (a_0 \cdot a_3)$

Infine $B_{min}^0 = 0$

