Esercizio n. 1

Data la seguente espressione logica:
a’*((c’+d’)’+e)+ ab*(e+ed+cd) + (cde’+e)’ *(a’+ab)

la si semplifichi, utilizzando le proprietà dell’algebra di commutazione. Riportare per ogni passaggio la proprietà utilizzata.

Soluzione:

a’*((c’+d’)’+e)+ ab*(e+ed+cd) + (cde’+e)’ *(a’+ab)
=> Assorbimento: e+ed = e

a’*((c’+d’)’+e)+ ab*(e+cd) + (cde’+e)’ *(a’+ab)
=> De Morgan: (c’+d’)’ = (c’)’*(d’)’;

a’*((c’)’*(d’)’+e)+ ab*(e+cd) + (cde’+e)’ *(a’+ab)
=> Involuzione: (c’)’ = c

a’*(c*(d’)’+e)+ ab*(e+cd) + (cde’+e)’ *(a’+ab)
=> Involuzione: (d’)’ = d

a’*(cd+e)+ ab*(e+cd) + (cde’+e)’ *(a’+ab)
=> Semplificazione: cde’+e = cd + e;

a’*(cd+e)+ ab*(e+cd) + (cd+e)’ *(a’+ab)
=> Commutativa: e + cd = cd +e;

a’*(cd+e)+ ab(cd+e) + (cd+e)’ (a’+ab)
=> Distributiva: a’(cd+e)+ ab(cd+e) = (cd+e)(a’+ab) ;

(cd+e)*(a’+ab) + (cd+e)’*(a’+ab)
=> Distributiva: (cd+e)*(a’+ab) + (cd+e)’*(a’+ab) = (a’+ab)*((cd+e)+ (cd+e)’);

(a’+ab)*((cd+e)+ (cd+e)’)
=> Inverso: (cd+e)+ (cd+e)’ = 1;

(a’+ab)*1
=> Elemento Neutro: (a’+ab)*1 = a’+ab;

a’+ab
=> Semplificazione: a’ + ab = a’ + b;

a’+b
=> Soluzione Finale.

Con Soluzioni

Esercizio n. 2

Data la seguente funzione ad una uscita, non completamente specificata:

F(a,b,c,d)= ONset(2,3,6,9,12,13,15) DCset(0,4,7)

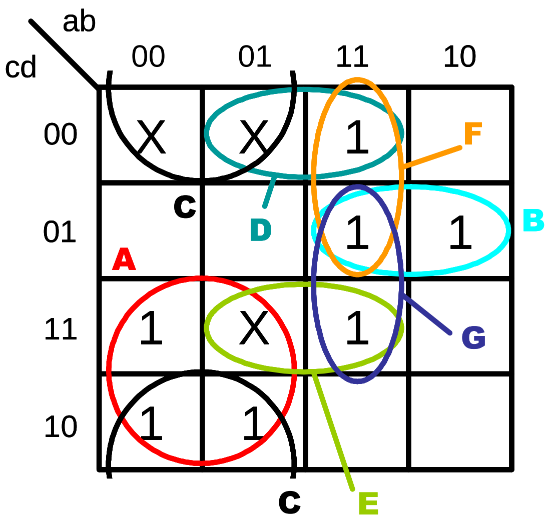
I) Sulla mappa di Karnaugh individuare gli implicanti primi **riportandone la forma algebrica** e separando gli implicanti *primi* da quelli *primi ed essenziali*.

II) Ricavare tutte le forme minime scegliendo una opportuna copertura della funzione sulla mappa, che minimizzi il numero di implicanti utilizzati ed il numero di letterali.

III) Ricavare il costo della copertura ottenuta, utilizzando come costo il numero di letterali.

Soluzione:

I)



Essenziali:
A = a'c
B = ac'd

Primi:
C = a'd'
D = bc'd'
E = bcd
F = abc'
G = abd

II)

A+B+D+E; A+B+D+G; A+B+E+F; A+B+F+G

III)

- Le soluzioni costano 2(A)+3(B)+3(D o F)+3(E o G)=11.

Esercizio n. 3

Data la seguente tabella di copertura:

	F1					F2					
	m0	m1	m2	m3	m4	m5	m6	m7	m8	m9	Costo
A					X				X		3
B						X	X	X			3
C			X						X		3
D	X	X					X	X			3
E										X	3
F						X				X	3
G								X			2
H			X	X							2
I					X						2
L	X	X									2
M				X							2

- Si trovi una copertura minima utilizzando il metodo di Quine McCluskey (m_{Xn} rappresenta un generico mintermine).
- Descrivere ogni singolo passo svolto per arrivare alla soluzione nella sequenza di applicazione

Soluzione:

F1=H+A+D

F2=F+A+D

PASSI:

1) F domina E -> E eliminato -> F essenziale per F2
2) H domina M -> M eliminato -> H essenziale per F1
3) A domina C -> C eliminato -> A essenziale per F2 -> costo A=1
4) A domina I -> I eliminato -> A essenziale per F2
5) M7 domina M6 -> M7 eliminato
6) D domina B -> B eliminato -> D essenziale per F2 -> costo D=1
7) D domina L -> L eliminato -> D essenziale per F1

Esercizio n. 4

Eseguire la generazione degli implicanti primi con il metodo di Quine McCluskey per la seguente funzione multiuscita F(F1;F2).

F1= on-set(m0, m5, m7, m12, m13)
dc-set(m4,m10,m11)

F2= on-set(m2,m3,m5,m7,m12)
dc-set(m8)

Soluzione:

Rilasso il problema trasformando il DC set in ON-Set

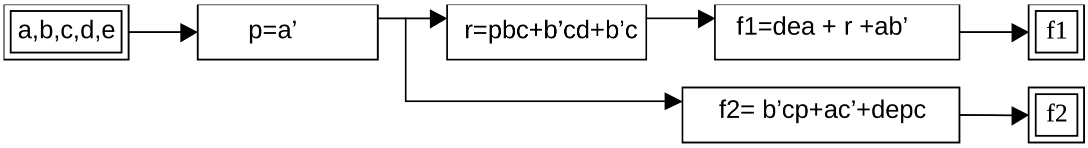
```
m0    0000  10    V
--
m2    0010  01    V
m4    0100  10    V
m8    1000  01    V
--
m3    0011  01    V
m5    0101  11    V
m10   1010  10    V
m12   1100  11    ( A )
--
m7    0111  11    V
m11   1011  10    V
m13   1101  10    V


m0m4       0-00  10    ( B )
--
m2m3       001-  01    ( C )
m4m5       010-  10    V
m4m12      -100  10    V
m8m12      1-00  01    ( D )
--
m3m7       0-11  01    ( E )
m5m7       01-1  11    ( F )
m5m13      -101  10    V
m10m11     101-  10    ( G )
m12m13     110-  10    V


m4m5m12m13     -10-  10     ( H )
```

Ritorno al problema iniziale, gli implicanti primi rimangono:

A, B, C, D, E, F, H

G copre solo DC di f1 quindi non è un implicante primo per il problema iniziale.

Esercizio n. 5



Data la rete multilivello sopra riportata, applicare in sequenza le trasformazioni sotto indicate e rispondere alle domande dove richiesto. Disegnare anche il modello della rete finale.

Nota Bene: per ogni trasformazione **è obbligatorio** riportare il **risultato della trasformazione** e **mostrare chiaramente tutti i passaggi** effettuati per ottenere il risultato stesso.

1. **COST( ):** Calcolo del numero di letterali. La funzione COST() calcola il costo in letterali indipendentemente dalla forma (SOP o Multilivello) delle espressioni algebriche dei nodi.

2. **SWEEP:** Eliminazione dei nodi costituiti da un solo letterale.
   2a. Mostrare <u>formalmente</u> che il costo della rete ottenuta applicando tale trasformazione è non peggiorativo.

3. **SIMPLIFY(r):** Minimizzazione a due livelli di r.
   3a. Mostrare <u>formalmente</u> che il costo della rete ottenuta applicando tale trasformazione al nodo r è non peggiorativo.

4. **ELIMINATE(r,-2):** Eliminazione vincolata del nodo r. Il parametro –2 indica la soglia di incremento di area per accettare o meno la trasformazione.

5. **FACTOR(f1):** Fattorizzazione del nodo f1.

6. **[s] = EXTRACT(f1,f2):** Estrazione di un fattore comune a f1 e f2. Il nodo s derivato dall'estrazione può essere un nuovo nodo o un nodo già presente nella rete.

7. **COST():** Calcolo del numero di letterali.

Soluzione

1. **COST( ):** 24 letterali

2. **SWEEP:** Viene eliminato il solo nodo p. Quindi:

r = a'bc+b'cd + b'c
f2= b'ca' + ac' + dea'c

2a. L'espressione (n*l – n – l) fornisce l'incremento di area in letterali di una rete a seguito dell'eliminazione di un nodo (l è il numero di letterali del nodo eliminato e n è il numero di nodi che lo assorbono). Nel caso di nodi eliminati costituiti da un solo letterale l'incremento di area è sempre pari a –1, qualunque sia il numero di nodi che assorbono.

3. **SIMPLIFY(r):** Minimizzazione a due livelli di r.

Tramite mappe di Karnaugh o manipolazione algebrica ottima, il risultato della minimizzazione dell'espressione r = a'bc+b'cd + b'c è

r = a'bc+b'c
r = c(a'b+b')
r = c(a'+b')
r = ca'+cb'

3a. L'espressione da minimizzare a due livelli è già una forma SOP, quindi la sua ottimizzazione non può essere peggiorativa (da SOP a SOP minima).

4. **ELIMINATE(r,-2):** Eliminazione vincolata del nodo r. Il parametro –2 indica la soglia di incremento di area per accettare o meno la trasformazione.

Applicando ancora l'espressione per il calcolo di incremento di area n*l – n – l (con l=4, numero di letterali di r e n=1, un solo nodo –f1- assorbe r), l'incremento risulta = -1. E' quindi al di sopra del valore –2 della soglia di accettazione. La trasformazione non viene accettata e le espressioni dei nodi restano quelle del passo precedente.

Lo stesso risultato si poteva ottenere eliminando il nodo e calcolando il nuovo costo della rete.

5. **FACTOR(f1):** Fattorizzazione del nodo f1.

L'algoritmo visto a lezione porta alla fattorizzazione

f1=a(de+b') + r

6. **[s] = EXTRACT(f1,f2):** estrazione di un fattore comune a f1 e f2. Il nodo s derivato dall'estrazione può essere un nuovo nodo o un nodo già presente nella rete.

s= de + b'
f1= as + r
f2= a'cs + ac'

7. **COST():** 15 letterali

La rete è infatti composta dai seguenti nodi

r = bc+b'c
s= de + b'
f1= as + r
f2= a'cs + ac'

Esercizio n. 6

Dati due numeri decimali A=0.546875 e B=2.1875. Fornire la codifica completa in virgola mobile a singola precisione di A e B.

Effettuare la somma A+B indicando tutti i passaggi relativi sia alla codifica che alla somma.

Soluzione:

A = 0.546875 = 0 01111110 00011000000000000000000

B = 2.1875 = 0 10000000 00011000000000000000000

Denormalizzo e sommo (NOTA: nella somma, il risultato ha l'esponente uguale all'esponente più alto dei due numeri, quindi si calcola la differenza dei due e si *porta* il numero a esponente più piccolo al pari del secondo e si *adatta* la mantissa. NOTA2: non avrebbe senso tenere come esponente del risultato quello del numero più piccolo!):

0.01000110000000000000000 +
1.00011000000000000000000 =
1.01011110000000000000000

La codifica normalizzata è:
2.734375 = 0 10000000 01011110000000000000000

Esercizio n. 8

Sia data una rete combinatoria con ingressi (a, b, c, d, e, f) e uscite (Y1, Y2, Y3) rappresentata dalla rete multilivello costituita dai seguenti nodi:

$$V_1 = \overline{a}bd + f$$

$$V_2 = a\overline{V_1}b + aV_1c + V_1cd + \overline{V_1}bd$$

$$V_3 = \overline{b}de + ab\overline{c}e + \overline{(\overline{b} + \overline{d} + \overline{e})} + a\overline{b}\overline{c}e + \overline{b}def$$

$$V_4 = \overline{a}b + bf(be + \overline{b})$$

$$Y_1 = V_4 + a\overline{f}$$

$$Y_2 = \overline{V_3}ab + \overline{a}bcd\overline{e} + \overline{b}cde + \overline{V_3}a\overline{b}f + \overline{a}bcde$$

$$Y_3 = a\overline{b}\overline{f} + d\overline{b}\overline{f} + ae + a\overline{V_2} + de + d\overline{V_2}$$

Applicare in sequenza alla rete multilivello le trasformazioni sotto indicate e rispondere alle domande dove richiesto. Disegnare anche il modello della rete finale.
Nota Bene: per ogni trasformazione **è obbligatorio** riportare il **risultato della trasformazione** e **mostrare chiaramente tutti i passaggi** effettuati per ottenere il risultato stesso.

a) **COST( ):** Calcolo del numero di letterali. La funzione COST() calcola il costo in letterali indipendentemente dalla forma (SOP o Multilivello) delle espressioni algebriche dei nodi.
b) **SIMPLIFY(Y_2):** Minimizzazione a due livelli di Y_2.
c) **SIMPLIFY(V_4):** Minimizzazione a due livelli di V_4.
d) **SIMPLIFY(V_3):** Minimizzazione a due livelli di V_3.
e) **ELIMINATE(V_4):** Eliminazione del nodo V_4
f) **FACTOR(V_2):** Fattorizzazione del nodo V_2.
g) **COST( ):** Calcolo del numero di letterali.

Soluzione:

a) COST( ):

b) **SIMPLIFY(Y_2):** Minimizzazione a due livelli di Y_2. Tramite mappe di Karnaugh o manipolazione algebrica ottima, il risultato della minimizzazione è

$$Y_2 = \bar{V}_3 ab + \bar{a}bcd\bar{e} + \bar{b}cde + \bar{V}_3 a\bar{b}f + \bar{a}bcde$$

$$Y_2 = \bar{V}_3 a(b + \bar{b}f) + \bar{a}bcd(\bar{e} + e) + \bar{b}cde$$

$$Y_2 = \bar{V}_3 a(b + f) + \bar{a}bcd + \bar{b}cde$$

$$Y_2 = \bar{V}_3 ab + \bar{V}_3 af + \bar{a}bcd + \bar{b}cde$$

c) **SIMPLIFY(V_4):** Minimizzazione a due livelli di V_4. il risultato della minimizzazione è

$$V_4 = \bar{a}b + bf(be + \bar{b})$$

$$V_4 = \bar{a}b + bf(e + \bar{b})$$

$$V_4 = \bar{a}b + bfe + bf\bar{b}$$

$$V_4 = \bar{a}b + bfe$$

d) **SIMPLIFY(V_3):** Minimizzazione a due livelli di V_3. il risultato della minimizzazione è

$$V_3 = \bar{b}de + ab\bar{c}e + \overline{(\bar{b} + \bar{d} + \bar{e})} + a\bar{b}\bar{c}e + \bar{b}def$$

$$V_3 = \bar{b}de + a\bar{c}e(b + \bar{b}) + bde + \bar{b}def$$

$$V_3 = \bar{b}de(1 + f) + a\bar{c}e + bde$$

$$V_3 = de(\bar{b} + b) + a\bar{c}e$$

$$V_3 = de + a\bar{c}e$$

e) **ELIMINATE(V_4):** l'unico nodo che contiene $\mathbf{V_4}$ è $\mathbf{Y_1}$

$$Y_1 = V_4 + a\overline{f}$$

$$Y_1 = \bar{a}b + bfe + a\overline{f}$$

f) **FACTOR(V_2):** Fattorizzazione del nodo V2.

$$V_2 = a\bar{V}_1 b + aV_1 c + V_1 cd + \bar{V}_1 bd$$

Il risultato della fattorizzazione è:

$$V_2 = (a + d)(\bar{V}_1 b + V_1 c)$$
