Esercizio 2 (7 punti) Si realizzi la sintesi di un circuito sequenziale che, presa in input una sequenza di bit, dia in output 1 ogni volta che negli ultimi tre bit sono contenuti due 0. Non si considerino sovrapposizioni.

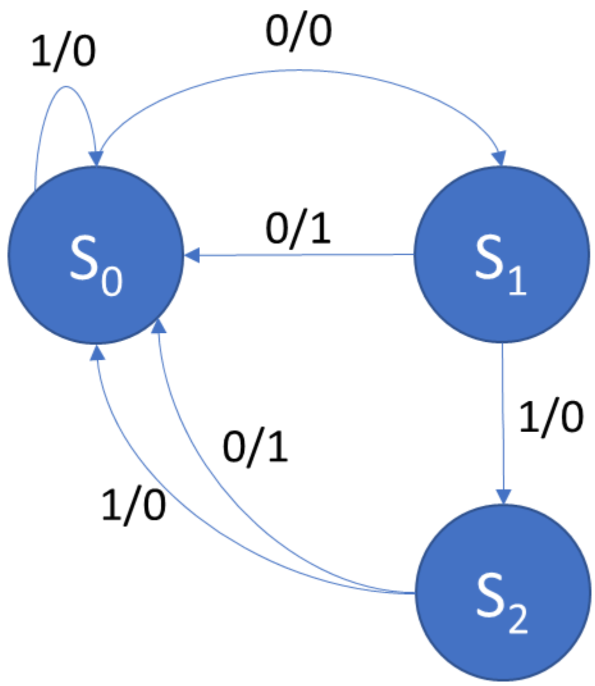
ESEMPIO IN: 1101’0100’11

OUT: 0000’1001’00

Si utilizzino FF di tipo SR per il bit di stato più significativo e FF di tipo T per i restanti bit.

Svolgimento

La macchina a stati (di Mealy) è la seguente:



La tavella della verità è:

PS,x	NS,out
S_0,0	S_1 , 0
S_0,1	S_0 , 0
S_1,0	S_0 , 1
S_1,1	S_2 , 0
S_2,0	S_0 , 1
S_2,1	S_0 , 0

Usando la codifica:

S_0,00
S_1,01
S_2,10

Posso scrivere:

S_1S_0,x	NS,out
000	01, 0
001	00, 0
010	00, 1
011	10, 0
100	00, 1
101	00, 0
11x	xx, x

Da cui

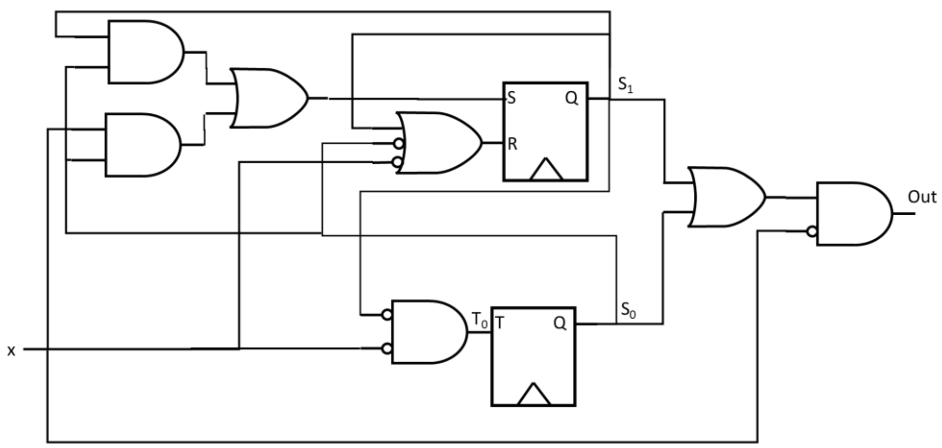
out = $\overline{S_1}S_0\overline{x} + S_1\overline{x} = S_0\overline{x} + S_1\overline{x}$

Per i FF di tipo T ed SR ho la seguente tabella:

S_1S_0,x	NS,out	Set_1	$Reset_1$	T_0
000	01, 0	0	x	1
001	00, 0	0	x	0
010	00, 1	0	x	1
011	10, 0	1	0	0
100	00, 1	0	1	0
101	00, 0	0	1	0
11x	xx, x	x	x	0

Da cui ricavo

$Set_1 = \overline{S_1}S_0x + S_1S_0 = S_0x + S_1S_0$
$Reset_1 = S_1+\overline{S_0}+\overline{x}$
$T_0 = \overline{S_1}\,\overline{x}$



Esercizio 3 (1+2+1 punti)

a) Rappresentare X = - 97 e Y = 39 in Ca2, ognuno con il minimo numero di bit.
b) Dopo aver calcolato il numero di bit necessario per rappresentare sia la somma X+Y che la differenza X-Y, portare X e Y alla lunghezza necessaria ed eseguire le due operazioni.
c) Infine, verificare i risultati ottenuti.

a)

X = -97 → -(97) → -(64 +32+1) → -(2^6+2^5+2^0) → -(01100001) → 10011110+1 → 1001_1111

Y = 39 → 32+4+2+1 → 2^5+2^2+2^1+2^0 → 100111

b)

A=X+Y = -58.

Un numero in complemento a 2 di N bits rappresenta i numeri nell'intervallo [-2^{N-1}: 2^{N-1}-1]. Con N=7 rappresentiamo i numeri nell'intervallo [-64:63].

L'operazione X+Y in binario è:

```
X   1001_1111
            +
Y   0010_0111 =
    ---------
    1100_0110
```

Convertiamo A per controllare che il valore sia corretto:
A= 1100_0110 =- (0011_1001+1) → -(0011_1010) → -(32+16+8+2) → -58

B=X-Y = -136. N=9 intervallo di rappresentazione [-256:255].

Quindi per l'operazione in binario utilizziamo 9 bits, sia per X, sia per -Y.

Per X estendiamo il segno, scrivendo X= 1_1001_1111
Poi Calcoliamo -Y.

Y = 0_0010_0111.

-Y = -(0_0010_0111) → 1_1101_1000+1 → 1_1101_1001

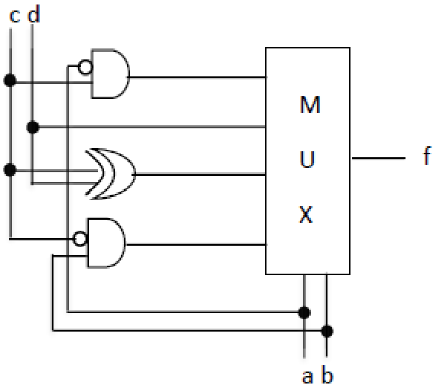
L'operazione X-Y in binario è:

```
X    1_1001_1111
               +
-Y   1_1101_1001 =
     -----------
B    1_0111_1000
```

Convertiamo B per controllare che il valore sia corretto:
B= 1_0111_1000 = -(0_1000_0111+1) → -(0_1000_1000) → -(128+8) →- 136

Esercizio 5 (2+2+1+2 punti)

- Si consideri il seguente circuito e si scrivano le espressioni della funzione f realizzata e della sua duale $\tilde{f}$.
- Trasformare poi tali espressioni, usando assiomi e regole dell'algebra di Boole, in forma normale (SOP o POS, a scelta)
- Si stenda la tavola di verità di f e di $\tilde{f}$.
- Si scrivano l'espressione minimale POS di f, l'espressione minimale SOP di $\tilde{f}$ e l'espressione canonica POS di $f \oplus \tilde{f}$.



a)

f si può ricavare dall'equazione del MUX.

Supponendo che i bit di selezione siano s_1 ed s_0 ed i bit di ingresso siano d_3 d_2 d_1 d_0 sappiamo che

$f = d_0\overline{s_1}\,\overline{s_0} + d_1\overline{s_1}\,s_0 + d_2 s_1\overline{s_0} + d_3 s_1 s_0$

Quindi possiamo scrivere

$f = (c\overline{a})\overline{a}\,\overline{b} + d\overline{a}\,b + (c \oplus d)a\overline{b} + (\overline{c}\,b)ab = \overline{a}\,\overline{b}\,c + \overline{a}\,bd + a\overline{b}\,(c \oplus d) + ab\overline{c} =$

$= \overline{a}\,\overline{b}\,c + \overline{a}\,bd + a\overline{b}\,(c\overline{d} + \overline{c}\,d) + ab\overline{c} = \overline{a}\,\overline{b}\,c + \overline{a}\,bd + a\overline{b}\,c\overline{d} + a\overline{b}\,\overline{c}\,d + ab\overline{c}$

b)

$f = \overline{a}\,\overline{b}\,c + \overline{a}\,bd + a\overline{b}\,c\overline{d} + a\overline{b}\,\overline{c}\,d + ab\overline{c}$ (forma SOP di f)

e la funzione duale $\tilde{f}$ è uguale ad:

$\tilde{f} = (\overline{a} + \overline{b} + c)(\overline{a} + b + d)(a + \overline{b} + c + \overline{d})(a + \overline{b} + \overline{c} + d)(a + b + \overline{c})$ (forma POS di $\tilde{f}$)

c)

abcd	$f\ \tilde{f}$	$f \oplus \tilde{f}$
0000	0 1	1
0001	0 1	1
0010	1 0	1
0011	1 0	1
0100	0 1	1
0101	1 0	1
0110	0 0	0
0111	1 1	0
1000	0 0	0
1001	1 1	0
1010	1 0	1
1011	0 1	1
1100	1 0	1
1101	1 0	1
1110	0 1	1
1111	0 1	1

d) Realizziamo le mappe di Karnaugh per *f* ed $\tilde{f}$

f

CD \ AB	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	1	1	0	0
10	0	1	0	1

$\tilde{f}$

CD \ AB	00	01	11	10
00	1	1	0	0
01	1	0	1	0
11	0	0	1	1
10	0	1	1	0

La forma minima POS di f è

$f = (a+c)\,(a+b+d)\,(\bar{a}+\bar{c}+\bar{d})(\bar{a}+\bar{b}+\bar{c})$

La forma minima SOP di $\tilde{f}$ è

$\tilde{f} = \bar{a}\,\bar{b}\,\bar{c} + \bar{a}\,\bar{c}\,\bar{d} + bc\bar{d} + acd + abd$

La forma canonica POS di $f \oplus \tilde{f}$ è

$f \oplus \tilde{f} = (a+\bar{b}+\bar{c}+d)\,(a+\bar{b}+\bar{c}+\bar{d})\,(\bar{a}+b+c+d)(\bar{a}+b+c+\bar{d})$

Esercizio 6 (1+2+1 punti)

a) Dati i numeri in rappresentazione IEEE 754 X = 0xc0e40000 e Y = 0xbf580000, (a) rappresentarli in notazione decimale (approssimato ± 0,03), (b) eseguire l'operazione X+Y e (c) rappresentare il risultato sia in notazione decimale a virgola mobile (approssimato ± 0,03) e sia in esadecimale.

(a)

X= 0xc0e40000 → 1100_0000_1 110_0100_0000_0000_0000_0000

Segno s_x =1

Esponente e_x = 1000_0001 = 129

Mantissa m_x = 1.110_0100_0000_0000_0000_0000 = 1+0.5+0.25+0.03125 = 1.78125 ≅ 0.78

X= $-1.78125*2^{129-127} = -1.78125*2^2 \cong -1.78*4 = -7.12$

Y= 0xbf580000 → 1011_1111_0101_1000_0000_0000_0000_0000

Segno s_y =1

Esponente e_y= 0111_1110 = 126

Mantissa m_y = 1.101_1000_0000_0000_0000_0000 = 1+0.5+0.125+0.0625 = 1.6875 ≅ 1.68

Y= $-1.6875*2^{126-127} = -1.6875*2^{-1} \cong -1.68/2 = -0.84$

(b)

Somma di Z=X ed Y

- Allineo la Y, che ha esponente minore facendo scorrere a destra la mantissa di e_x -e_y posizioni (3 posizioni).

(m_y >>3) = 0001_1011_0000_0000_0000_0000

- Sommo m_x ed (m_y >>3)

m_x = 1.110_0100_0000_0000_0000_0000

(m_y >>3) = 0.001_1011_0000_0000_0000_0000

1.111_1111_0000_0000_0000_0000

Quindi

m_z =1.111_1111_0000_0000_0000_0000 = (1+0.5+0.25+0.125+0.0625+0.03125…) ≅ 1.96

e_z corrisponde all'esponente massimo tra e_x ed e_y. Quindi

e_z =max(e_x, e_y)=129

e s_z =1.

Possiamo quindi costruire la rappresentazione IEEE 754 di Z come

1 1000_0001 111_1111_0000_0000_0000_0000

In esadecimale

Z = c0ff0000

In notazione decimale

$Z \cong -1.96*4 \cong -7.84$

Notate che a causa nell'approssimazione $m_z \cong -1.96$ il valore di Z è diverso dalla somma -7.12-0.84= -7.96.

Se avessimo usato un valore più preciso di $m_z = 1.9921875$ otterremmo il valore esatto di Z= 7.96875.