Appello di Progettazione di Sistemi Digitali
16 Settembre 2013 - Docenti: Proff. Gorla e Massini

Esercizio 1 (3 punti): Convertire in base 4 con rappresentazione in virgola fissa il numero decimale 214,1362 avendo a disposizione 5 cifre per la parte intera e 6 per la parte decimale. La rappresentazione ottenuta è precisa o è un'approssimazione del numero decimale di partenza?

SOLUZIONE:

Convertiamo la parte intera, col metodo delle divisioni ripetute:

214 : 4 = 53 resto 2
53 : 4 = 13 resto 1
13 : 4 = 3 resto 1
3 : 4 = 0 resto 3

Convertiamo la parte decimale, col metodo delle moltiplicazioni ripetute:

0,1362 x 4 = 0,5448
0,5448 x 4 = 2,1792
0,1792 x 4 = 0,7168
0,7168 x 4 = 2,8672
0,8672 x 4 = 3,4688
0,4688 x 4 = 1,8752

Quindi, il numero cercato, nella rappresentazione richiesta è 03112,020231
Questa è una rappresentazione approssimata, visto che la conversione della parte decimale non è terminata con un valore pari a 0.

Esercizio 2 (4 punti): Dati i seguenti valori nella rappresentazione in virgola mobile con base 2 ed esponente in complemento a 2, eseguire la somma e mostrare il risultato sotto forma di tripla normalizzata, utilizzando 10 bit per la mantissa e 5 per l'esponente.

< 1 , 11100100 , 1011 >   < 0 , 11101000 , 1101 >

SOLUZIONE:

Il primo numero ha esponente pari a -5, il secondo a -3; bisogna pertanto portare il primo numero (quello con esponente più basso) all'esponente del secondo. Il primo numero diventa pertanto:

< 1 , 0011100100 , 1101 >

Visto che il primo numero ha una mantissa minore di quella del secondo, e visto che il primo numero è negativo, bisogna ora effettuare la differenza tra la mantissa del secondo numero e la mantissa del primo:

```
    1110100000 –
    0011100100 =
----------------------------------
    1010111100
```

da cui il risultato (già normalizzato) è < 0 , 1010111100 , 11101 >. Si noti che, per adottare il formato richiesto, è stato aggiunto un 1 all'inizio dell'esponente, per farlo passare da numero in complemento a due da 4 bit a numero in complemento a due da 5 bit.

Esercizio 3 (2+1+1 punti): Utilizzando gli assiomi dell'algebra di Boole, si dimostri che

$$\bar{a} + bc = \bar{a} + c(\bar{a} + b) + abc$$

Si scrivano poi la duale e la complementare di tale uguaglianza.

SOLUZIONE:

$\underline{a} + c(\underline{a} + b) + abc$	=	$\underline{a} + c\underline{a} + cb + abc$	(distrributività)
	=	$\underline{a}(1 + c) + bc(1 + a)$	(elem.neutro, commutat., distrributiv.)
	=	$\underline{a}(1) + bc(1)$	(elem.annullatore)
	=	$\underline{a} + bc$	(elem.neutro)

DUALE: $\underline{a}(b + c) = \underline{a}(c + \underline{a}b)(a + b + c)$

COMPLEMENTARE: $a(\underline{b} + \underline{c}) = a(\underline{c} + a\underline{b})(\underline{a} + \underline{b} + \underline{c})$

Esercizio 4 (2+2+3 punti): Una funzione di 4 variabili, f(x4,x3,x2,x1), vale 1 se x4 + x2 = 0, risulta non specificata (termini don't care) se si verifica la condizione x3x2 = 1 e vale 0 in ogni altro caso. Calcolare con le mappe di Karnaugh l'espressione POS minimale e da questa progettare la rete che realizza la funzione tramite **multiplexer** con **3 variabili** di selezione.

SOLUZIONE:

Iniziamo a mettere 1 in tutte le righe in cui sia x4 che x2 valgono 0; poi mettiamo - in tutte le righe in cui sia x3 che x2 valgono 1; infine mettiamo 0 in tutte le righe restanti. Si ottiene così la tabella per f:

x4	x3	x2	x1	f
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	-
0	1	1	1	-
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	-
1	1	1	1	-

Costruiamo la mappa di Karnaugh e copriamo massimamente gli 0 e, se necessario, i don't care::

x2x1 \ x4x3	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	0	-	-	0
10	0	-	-	0

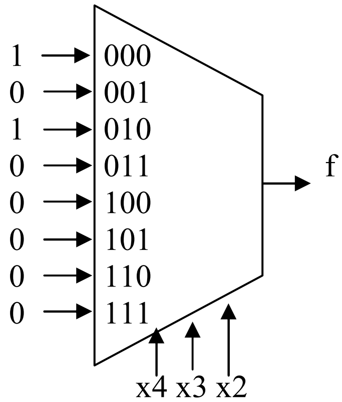
I maxtermini corrispondenti ai ricoprimenti mostrati sono:

per il rosso: $\underline{x4}$

per il blu: $\underline{x2}$

Pertanto, la POS minimale è $\underline{x4}\cdot\underline{x2}$

Per realizzare tale espressione con un MUX 8-a-1 (con 3 segnali di controllo), conviene usare come controlli sia x4 che x2; il terzo controllo può essere scelto a piacere, per esempio x3. Il circuito richiesto è:



Esercizio 5 (2+2+2 punti): Disegnare l'automa di Mealy che, presa in input una sequenza di bit, dà in output 1 se e solo se l'AND logico degli ultimi due bit di indice dispari ricevuti fino a quel momento è 1 (si consideri che il primo bit ricevuto ha indice 1). Ad esempio:

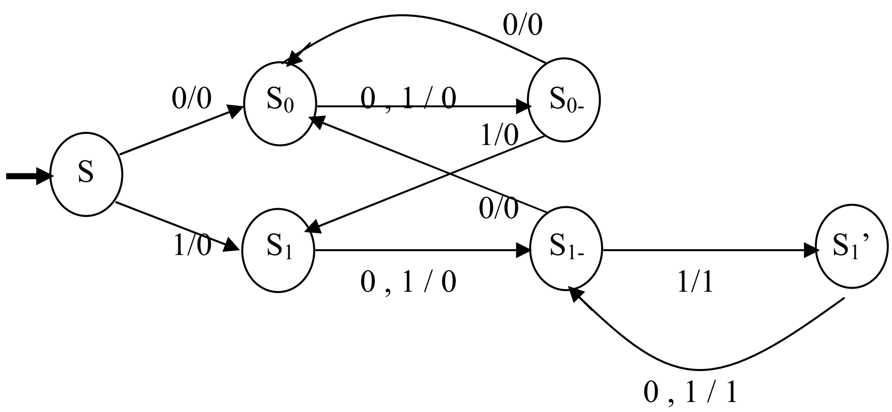
INPUT: 001011101001

OUTPUT: 000011111100

Si minimizzi l'automa dato e, dall'automa minimo, si ricavi l'automa di Moore equivalente.

SOLUZIONE:

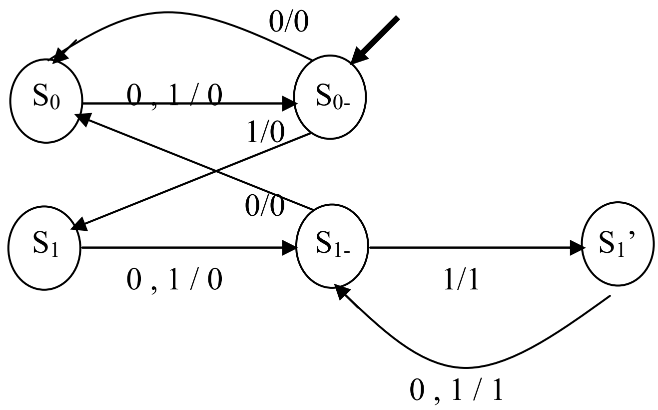
Dallo stato iniziale S si può ricevere 0 o 1, andando rispettivamente negli stati S_0 ed S_1, corrispondenti al fatto che abbiamo ricevuto un bit di indice dispari e tale bit è 0 o 1. Qualsiasi cosa riceviamo (il bit seguente sarà di indice pari e quindi da ignorare), andiamo negli stati S_{0-} ed S_{1-}, corrispondenti al fatto che abbiamo ricevuto un bit pari e l'ultimo bit dispari era 0 o 1. Fino a qui tutti gli output sono 0. Dallo stato S_{0-} transito nello stato S_0 o S1, a seconda che ricevo 0 o 1, dando in entrambe i casi 0 perché devo fare l'AND tra 0 e un altro bit. S_{1-} si comporta in modo diverso: se riceve 0, va in S_{0-} e dà 0; se riceve 1, dà 1 e va in un nuovo stato S_1'. Da tale stato, qualunque cosa riceve, va in S_{1-} dando in output 1, visto che l'AND degli ultimi due bit di indice dispari è 1. Pertanto:



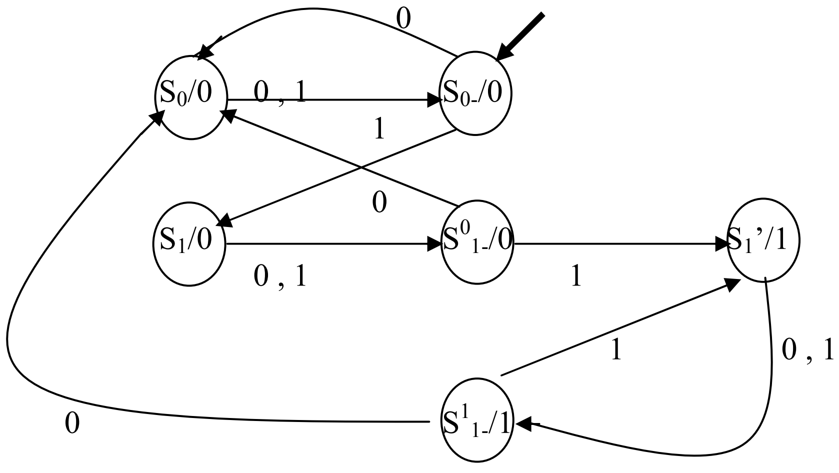
Per minimizzare l'automa, si utilizza la tabella triangolare:

S_0	X				
S_1	X	X			
S_{0-}		X	X		
S_{1-}	X	X	X	X	
S_1'	X	X	X	X	X
	S	S_0	S_1	S_{0-}	S_{1-}

da cui si evince che S può essere fuso con S_{0-}. Pertanto l'automa di Mealy minimo è:



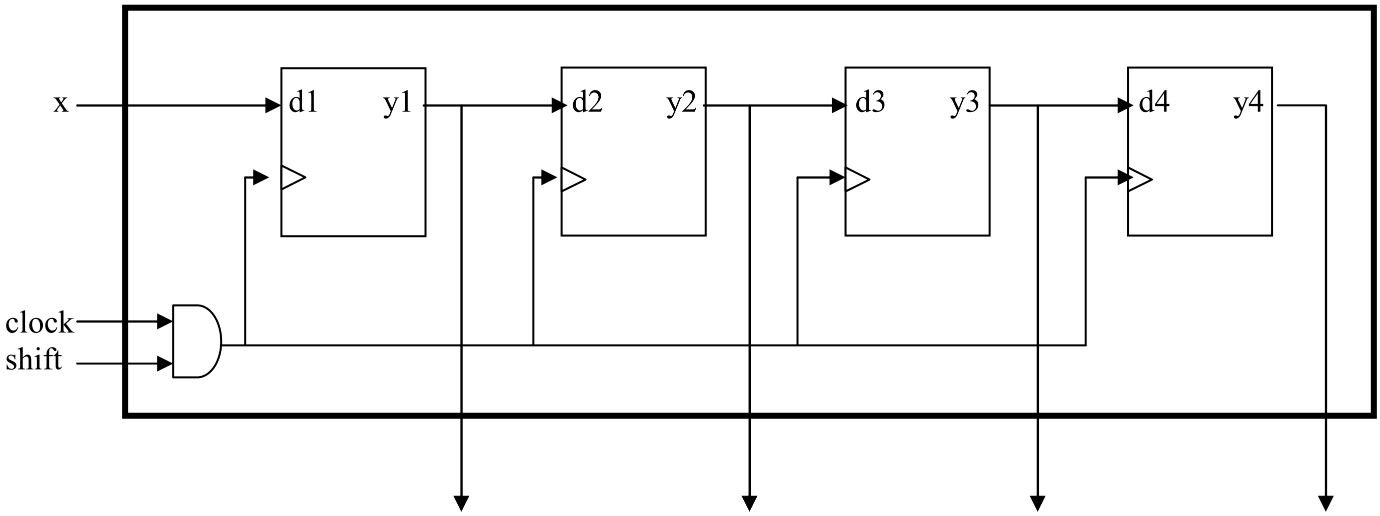
Per passare all'automa di Moore equivalente, basta notare che solo S_{1-} ha transizioni entranti con output diverso. Lo scomponiamo quindi in due stati, S^0_{1-} ed S^1_{1-} e otteniamo:



Esercizio 6 (3+3 punti): Si disegni un registro SIPO da 4 bit costruito utilizzando flip-flop di tipo D. Si mostri poi il diagramma temporale con la sequenza di input 0100.

SOLUZIONE:

Il registro a input seriale e output parallelo (SIPO) realizzato mediante FF di tipo D è il seguente:



Per disegnare il diagramma temporale, assumiamo che il segnale di shift sia uguale al segnale di clock e l'input (x) commuti ad ogni segnale di clock. Assumiamo anche che i FF siano inizialmente tutti a 0. Pertanto avremo: