

**PRIMA PROVA INTERMEDIA DEL CORSO DI
CALCOLATORI ELETTRONICI
NUOVO ORDINAMENTO DIDATTICO**

15 Aprile 2009

NOME:

COGNOME:

MATRICOLA:

ESERCIZIO 1 (11 punti)

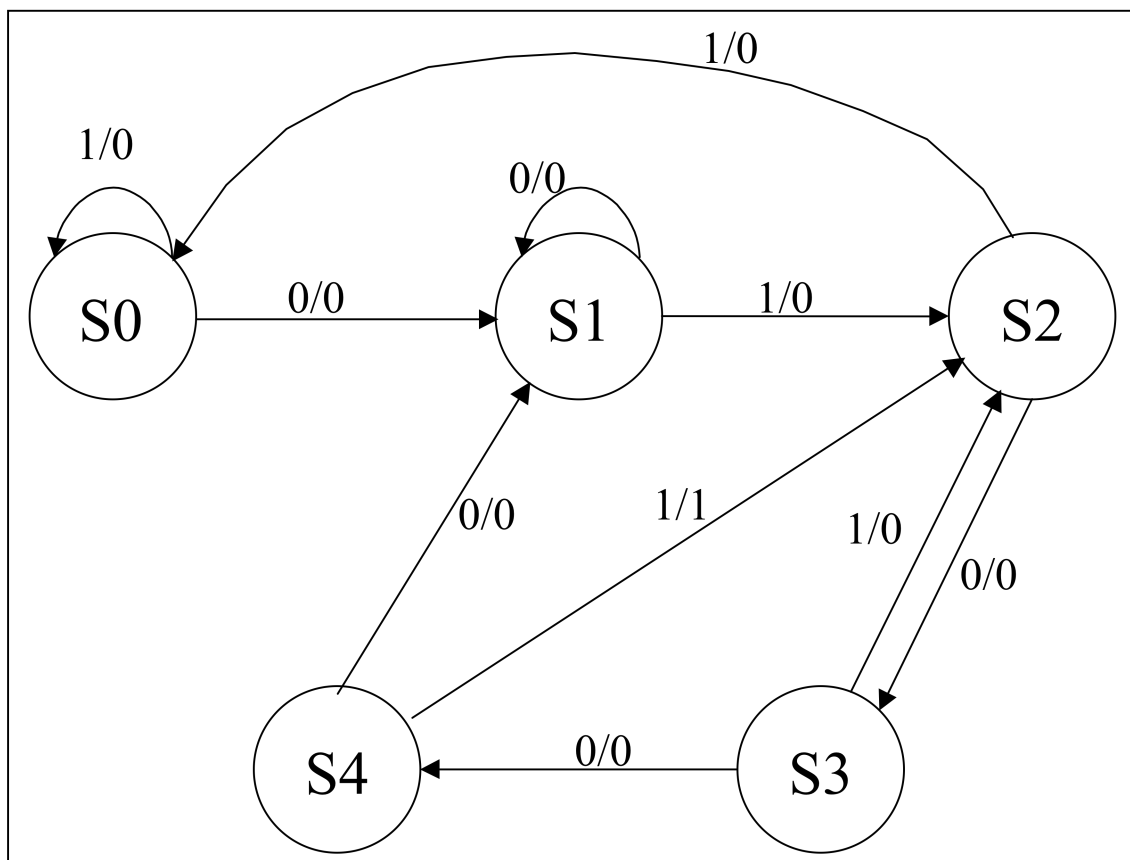
Progettare una rete sequenziale che presenti un ingresso X e un'uscita Z posta a 1 ogni volta che viene riconosciuta la sequenza 01001.

Si richiede:

- (4 punti) il diagramma degli stati, la tabella di flusso e la tabella delle transizioni;
- (3 punti) il calcolo delle forme minime delle variabili di eccitazione dei flip flop con le mappe di Karnaugh. Si usino flip flop JK. Sintetizzare anche la rete combinatoria per l'uscita Z.
- (2 punti) Spiegare come si possono sintetizzare le reti combinatorie per la transizione degli stati utilizzando un decoder opportuno ed una memoria ROM.
- (2 punti) Spiegare in che modo si potrebbe variare il circuito appena realizzato per riconoscere la sequenza 10110 (che è il complemento a 1 della sequenza originale).

Soluzione.

a) Il diagramma degli stati è il seguente:



La tabella di flusso è data da:

Stato presente	Stato successivo/Uscita	
	X=0	X=1
S0	S1/0	S0/0
S1	S1/0	S2/0
S2	S3/0	S0/0
S3	S4/0	S2/0
S4	S1/0	S2/1

Per codificare 5 stati occorrono tre flip flop. La codifica è la seguente:

S0 → 0 0 0; ...; S4 → 1 0 0. Nel seguito indicheremo ciascun bit della codifica con le lettere A, B, C. L'apice indicherà il bit nell'istante successivo a quello considerato.

A partire dalla tabella di eccitazione del flip flop JK:

Q	Q'	J	K
0	0	0	D
0	1	1	D
1	0	D	1
1	1	D	0

A	B	C	X	A'	Ja	Ka	B'	Jb	Kb	C'	Jc	Kc	Z
0	0	0	0	0	0	D	0	0	D	1	1	D	0
0	0	0	1	0	0	D	0	0	D	0	0	D	0
0	0	1	0	0	0	D	0	0	D	1	D	0	0
0	0	1	1	0	0	D	1	1	D	0	D	1	0
0	1	0	0	0	0	D	1	D	0	1	1	D	0
0	1	0	1	0	0	D	0	D	1	0	0	D	0
0	1	1	0	1	1	D	0	D	1	0	D	1	0
0	1	1	1	0	0	D	1	D	0	0	D	1	0
1	0	0	0	0	D	1	0	0	D	1	1	D	0
1	0	0	1	0	D	1	1	1	D	0	0	D	1
1	0	1	0	D	D	D	D	D	D	D	D	D	D
1	0	1	1	D	D	D	D	D	D	D	D	D	D
1	1	0	0	D	D	D	D	D	D	D	D	D	D
1	1	0	1	D	D	D	D	D	D	D	D	D	D
1	1	1	0	D	D	D	D	D	D	D	D	D	D
1	1	1	1	D	D	D	D	D	D	D	D	D	D

b) Ora possiamo disegnare le mappe di Karnaugh:

		AB			
		00	01	11	10
CX	00			d	d
	01			d	d
	11			d	d
	10		1	d	d

$$J_A = BC\bar{X}$$

		AB			
		00	01	11	10
CX	00	d	d	d	1
	01	d	d	d	1
	11	d	d	d	d
	10	d	d	d	d

$$K_A = 1$$

		AB			
		00	01	11	10
CX	00		d	d	
	01		d	d	1
	11	1	d	d	d
	10		d	d	d

$$J_B = CX + AX$$

		AB			
		00	01	11	10
CX	00	d		d	d
	01	d	1	d	d
	11	d		d	d
	10	d	1	d	d

$$K_B = \bar{C}X + C\bar{X}$$

		AB			
		00	01	11	10
CX	00	1	1	d	1
	01			d	
	11	d	d	d	d
	10	d	d	d	d

$$J_C = \bar{X}$$

		AB			
		00	01	11	10
CX	00	d	d	d	d
	01	d	d	d	d
	11	1	1	d	d
	10		1	d	d

$$K_C = B + X$$

Infine, per quanto riguarda l'uscita Z:

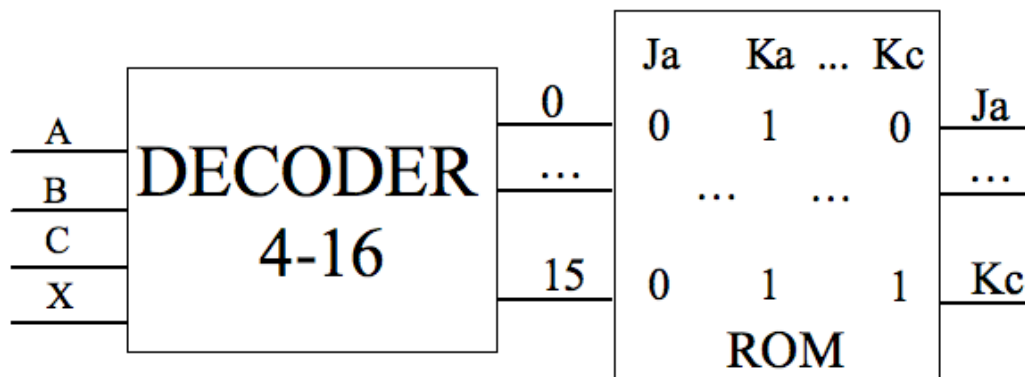
$$Z = A \cdot \overline{B} \cdot \overline{C} \cdot X$$

Volendo utilizzare anche i don't care:

AB \ CX		00	01	11	10
CX \ AB	00			d	
	01			d	1
	11			d	d
	10			d	d

$$Z = AX$$

c) Si può utilizzare un decoder 4-16 e interfacciarlo su una ROM, in cui siano memorizzati i valori di Ja, Ka, Jb, Kb, Jc e Kc, nel seguente modo:



NB: I valori segnati nella ROM sono coerenti con le funzioni booleane sintetizzate in forma minima (non posso memorizzare il valore D nella ROM).

d) E' sufficiente negare l'ingresso X della rete sequenziale: con una semplice porta not realizziamo il riconoscitore della sequenza 10110, partendo dal riconoscitore di sequenza 01001.

ESERCIZIO 2 (6 punti)

I trasferimenti di parole a/dalla memoria di un calcolatore sono codificati utilizzando il codice di Hamming. Si consideri la parola di 10 bit 0110101100 (il bit meno significativo è a sinistra).

Spiegando bene ogni passo del ragionamento:

- 1) (2 punti) calcolare il minimo numero di bit di controllo necessari per la codifica della parola;
- 2) (2 punti) codificare la parola data;
- 3) (2 punti) imporre un errore nel quinto bit della **parola inizialmente data**. Spiegare come l'errore viene rivelato e corretto per mezzo della codifica di Hamming.

Soluzione.

- 1) Deve venire rispettata la condizione:

$$2^K \geq N + K + 1 \quad (1),$$

dove K è il numero di bit di controllo inseriti. Essendo N=10, il numero minimo di bit di controllo richiesto è 4.

- 2) Nella codifica di Hamming, la sequenza in ingresso presenta la seguente struttura:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
c ₀	c ₁	b ₀	c ₂	b ₁	b ₂	b ₃	c ₃	b ₄	b ₅	b ₆	b ₇	b ₈	b ₉
		0		1	1	0		1	0	1	1	0	0

Dove c₀...c₃ sono i quattro bit costituenti il vettore di controllo, e b₀...b₉ i 10 bit trasmessi. Tali bit si ottengono con le seguenti operazioni

$$c_0 = b_0 \oplus b_1 \oplus b_3 \oplus b_4 \oplus b_6 \oplus b_8 = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 = 1$$

$$c_1 = b_0 \oplus b_2 \oplus b_3 \oplus b_5 \oplus b_6 \oplus b_9 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$c_2 = b_1 \oplus b_2 \oplus b_3 \oplus b_7 \oplus b_8 \oplus b_9 = 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$c_3 = b_4 \oplus b_5 \oplus b_6 \oplus b_7 \oplus b_8 \oplus b_9 = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 1$$

La stringa codificata è 10011101101100.

- 3) Nell'ipotesi di un errore sul quinto bit della stringa iniziale, la stringa ricevuta risulta: 10011101001100. Per rivelare questo errore, bisogna ricalcolare i bit di controllo:

$$c'_0 = b_0 \oplus b_1 \oplus b_3 \oplus b_4 \oplus b_6 \oplus b_8 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$c'_1 = b_0 \oplus b_2 \oplus b_3 \oplus b_5 \oplus b_6 \oplus b_9 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$c'_2 = b_1 \oplus b_2 \oplus b_3 \oplus b_7 \oplus b_8 \oplus b_9 = 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$c'_3 = b_4 \oplus b_5 \oplus b_6 \oplus b_7 \oplus b_8 \oplus b_9 = 0 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 0$$

Il passo successivo è calcolare il vettore di errore dato dalla differenza dei vettori di controllo c e c' (ricordiamo che somma e differenza tra bit producono lo stesso risultato):

$$e_0 = c_0 \oplus c'_0 = 1$$

$$e_1 = c_1 \oplus c'_1 = 0$$

$$e_2 = c_2 \oplus c'_2 = 0$$

$$e_3 = c_3 \oplus c'_3 = 1$$

Poiché il vettore risultante 1001 non è nullo, vi è un errore nella stringa di 14 bit e precisamente nella posizione indicata dal vettore di errore tradotto in notazione decimale (posizione 9). Il bit sbagliato nella stringa codificata è quindi b₄, che può essere corretto.

ESERCIZIO 3 (8 punti)

1. (6 punti) Si consideri una memoria primaria costituita da 1024 parole e una memoria cache costituita da 16 parole. Si consideri la seguente lista di chiamate (indirizzi espressi in decimale): 0, 1, 2, 3, 4, 22, 76, 77, 78, 2, 3, 4, 15, 2. Si indichi il contenuto della cache, ovvero quali parole occupano i relativi blocchi di cache, dopo l'ultima chiamata, e si calcoli il valore dell'hit rate, nel caso in cui:
 - a. (3 punti) Il metodo di indirizzamento della cache sia quello completamente associativo;
 - b. (3 punti) Il metodo di indirizzamento della cache sia quello associativo su insiemi a due vie con blocchi di 4 parole.La cache utilizza una strategia di rimpiazzamento FIFO.
2. (2 punti) Si consideri una gerarchia di memoria a tre livelli costituita da: cache, primaria e disco. Si consideri la cache descritta al punto precedente 1.a, con indirizzamento completamente associativo. Utilizzando l'hit rate della suddetta cache, calcolato al punto precedente e considerando che l'hit rate della primaria è pari a 0.99, i tempi di accesso a cache, primaria e disco valgono rispettivamente 4 ns, 40 ns e 5 ms, si esprima il tempo medio di accesso alla gerarchia **in nanosecondi**.

Soluzione.

1.a <TAG 8 bit><Offset 2 bit>

Blocco 0	Blocco 1	Blocco 2	Blocco 3
12	0	20	76
13	1	21	77
14	2	22	78
15	3	23	79

$$H = 8/14 = 0.57$$

1.b <TAG 7 bit><S.I. 1 bit><Offset 2 bit>

Set 0		Set 1	
Blocco 0	Blocco 1	Blocco 0	Blocco 1
0		12	4
1		13	5
2		14	6
3		15	7

$$H = 8/14 = 0.57$$

2. Con tutti i dati a nostra disposizione è sufficiente valutare la formula:

$$\begin{aligned}\bar{T} &= H_c T_c + (H_p - H_c)(T_p + T_c) + (1 - H_p)(T_d + T_p + T_c) \\ \bar{T} &= 50021.2ns\end{aligned}$$

ESERCIZIO 4 (8 punti)

1) (6 punti) Un disco presenta le seguenti caratteristiche: 6000 giri/min, 100 settori per traccia, tempo di posizionamento da una traccia a quelle adiacente 1 ms, 100 tracce per superficie, 32 B per settore.

Calcolare il tempo medio di lettura di un blocco di 1 KB da disco, nell'ipotesi che il primo settore utile si trovi nella prima traccia, che la testina sia pronta per trasferire il primo settore, e che i settori del blocco siano situati, a quattro a quattro, in tracce diverse distanti mediamente due tracce. I quattro settori presenti in ogni traccia sono memorizzati in modo contiguo.

2) (2 punti) Dopo un processo di deframmentazione del disco, i settori del blocco si trovano memorizzati in modo contiguo, tutti sulla stessa traccia. La testina è pronta per la lettura del primo blocco. Di quanto si riduce in percentuale il tempo di lettura del blocco in questo caso?

Soluzione.

1)

$$TROT = 60 / 6000 = 0.01 \text{ secondi}$$

$$TLAT = TROT / 2 = 0.005 \text{ secondi (tempo di latenza)} = 5 \text{ ms}$$

$$Tlett = TROT / 100 = 0.1 \text{ ms (tempo di lettura di un settore)}$$

$$Tsp = 1 \text{ ms}$$

$$Tpos = 2 * Tsp = 2 \text{ ms.}$$

$$\text{Numero di settori richiesti per il blocco da 1 KB: } 1024B/32B = 32.$$

Essendo i settori memorizzati in modo contiguo e a 4 a 4 per traccia, si deduce che il blocco da trasferire è memorizzato su 8 tracce diverse (32 settori / (4 settori/traccia)). Visto che la testina è pronta per trasferire il primo blocco, significa che si dovranno considerare solo $8-1 = 7$ tempi di latenza e posizionamento (quest'ultimo per via del fatto che i 4 settori memorizzati in ogni traccia sono contigui).

Tempo di lettura del blocco da 1KB:

$$= 32 * Tlett + 7 * (TLAT + Tpos) = 32 * 0.1 + 7 * 7 \text{ ms} = 52.2 \text{ ms}$$

2)

$$\text{In questo caso, banalmente } T = 32 * Tlett = 3.2 \text{ ms}$$

$$\text{La riduzione equivale quindi a } (52.2-3.2)/52.2 = 93.8\%$$