

SOLUZIONI DELLA PROVA SCRITTA DEL CORSO DI
CALCOLATORI ELETTRONICI
NUOVO ORDINAMENTO DIDATTICO

4 Ottobre 2001

MOTIVARE IN MANIERA CHIARA LE SOLUZIONI PROPOSTE A CIASCUNO DEGLI ESERCIZI SVOLTI

ESERCIZIO 1 (9 punti)

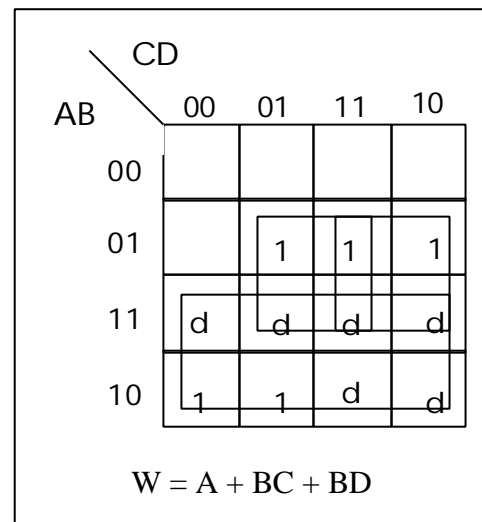
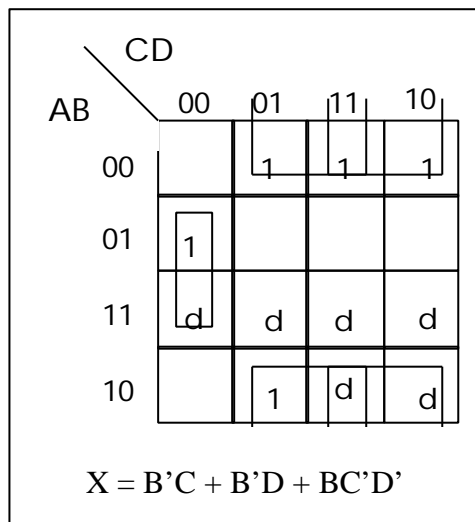
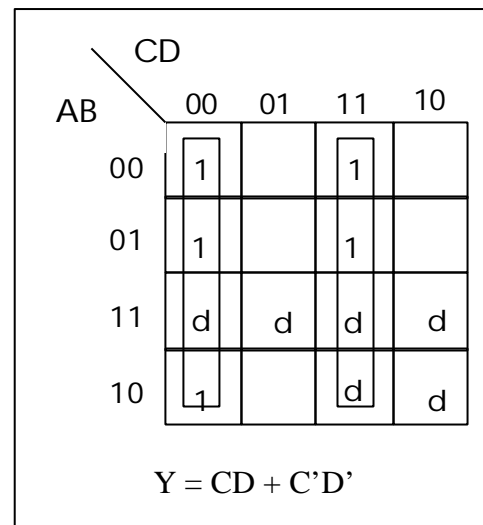
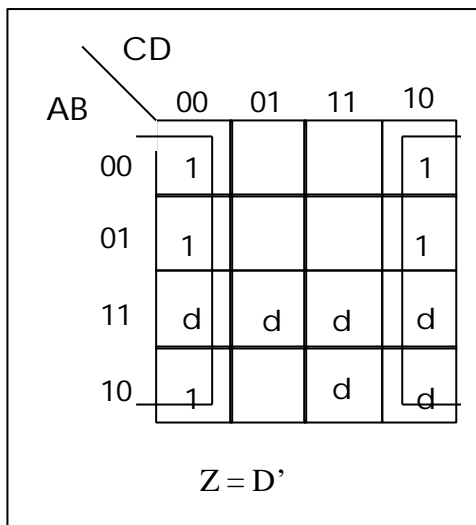
Progettare una rete combinatoria che effettui la conversione di un numero binario dal formato BCD (si indichino i quattro bit con A, B, C, D) al formato in eccesso 3 (indicando i bit con W, X, Y, Z). Individuare le forme minime per l'espressione di ciascun bit convertito con le mappe di Karnaugh.

Soluzione.

Se un numero è rappresentato in "eccesso 3", significa che il minimo valore assunto è l'eccesso 3, corrispondente alla combinazione per cui tutti i bit del numero BCD sono pari a zero. Quindi si può passare da un numero in formato BCD al corrispondente in eccesso 3 semplicemente sommando il valore 0011 (3 in base 10) al numero originale. Poiché utilizziamo 4 bit, necessari per il BCD, la corrispondenza è:

BCD				Eccesso 3			
A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

Le restanti combinazioni di quattro bit possono essere associate ad altrettanti "don't care". Possiamo ora scrivere la mappa relativa ad ogni nuovo bit W, X, Y, Z, in funzione dei quattro bit di input A, B, C, D, in formato BCD.



ESERCIZIO 2 (8 punti)

I trasferimenti di parole a/dalla memoria di un calcolatore sono codificate utilizzando il codice di Hamming. Si consideri la stringa di 8 bit 10110001 (il bit meno significativo è a sinistra). **Spiegando bene ogni passo del ragionamento:**

- 1) (1 punto) Calcolare il minimo numero di bit di controllo.
- 2) (3 punti) Codificare la stringa con il codice di Hamming.
- 3) (4 punti) Si supponga un errore nel nono bit della stringa codificata. Mostrare in che modo tale errore viene rilevato e corretto.

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 $N = 8$, si evince dalla (1) che il numero minimo di bit di controllo richiesto è 4.

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

c_0	c_1	b_0	c_2	b_1	b_2	b_3	c_3	b_4	b_5	b_6	b_7
		1		0	1	1		0	0	0	1

Dove $c_0 \dots c_3$ sono i quattro bit costituenti il vettore di controllo, e $b_0 \dots b_7$ gli otto bit trasmessi.

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

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

$$c_2 = b_1 \oplus b_2 \oplus b_3 \oplus b_7 = 1$$

$$c_3 = b_4 \oplus b_5 \oplus b_6 \oplus b_7 = 1$$

La stringa codificata è: 011101110001.

- 3) Per verificare la presenza di un errore nel sesto bit, dobbiamo ricalcolare il vettore di controllo a partire dalla sequenza ricevuta. Si ha:

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

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

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

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

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):

$$e_0 = c_0 \oplus c'_0 = 0$$

$$e_1 = c_1 \oplus c'_1 = 1$$

$$e_2 = c_2 \oplus c'_2 = 1$$

$$e_3 = c_3 \oplus c'_3 = 0$$

Poiché il vettore risultante 0110 non è nullo, vi è un errore nella stringa di 12 bit data e precisamente nella posizione indicata dal vettore di errore tradotto in notazione decimale. Il bit sbagliato è quindi il sesto (b2), come supposto.

ESERCIZIO 3 (7 punti)

La memoria di un calcolatore è gestita con una tecnica di 'paginazione su richiesta'. Si consideri la seguente richiesta di pagine:

8,1,1,2,8,1,3,9,2,4,3,6,9,7,6,1,7,0,3,9,7.

Se la memoria contiene complessivamente quattro pagine, calcolare l' hit ratio e il numero di 'page faults' nei seguenti due casi:

(a) strategia di rimpiazzamento delle pagine FIFO. (3 punti)

(b) strategia di rimpiazzamento delle pagine LRU. (4 punti)

Soluzione:

(a) Page trace nel caso di strategia di rimpiazzamento delle pagine FIFO (X = hit)

Tempo	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Richieste	8	1	1	2	8	1	3	9	2	4	3	6	9	7	6	1	7	0	3	9	7
Pagine	8	1	1	2	2	2	3	9	9	4	4	6	6	7	7	1	1	0	3	9	7
		8	8	1	1	1	2	3	3	9	9	4	4	6	6	7	7	1	0	3	9
				8	8	8	1	2	2	3	3	9	9	4	4	6	6	7	1	0	3
							8	1	1	2	2	3	3	9	9	4	4	6	7	1	0
Hit			X		X	X			X		X		X		X		X				

Hit ratio = 8/21; Page faults = 13/21

(b) Page trace nel caso di strategia di rimpiazzamento delle pagine LRU (X = hit)

Tempo	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Richieste	8	1	1	2	8	1	3	9	2	4	3	6	9	7	6	1	7	0	3	9	7
Pagine	8	1	1	2	8	1	3	9	2	4	3	6	9	7	6	1	7	6	3	9	7
		8	8	1	2	8	1	3	9	2	4	3	6	9	7	6	1	7	6	3	9
				8	1	2	8	1	3	9	2	4	3	6	9	7	6	1	7	6	3
							2	8	1	3	9	2	4	3	3	9	9	9	1	7	6
Hit			X		X	X					X				X		X	X			X

Hit ratio = 8/21; Page faults = 13/21

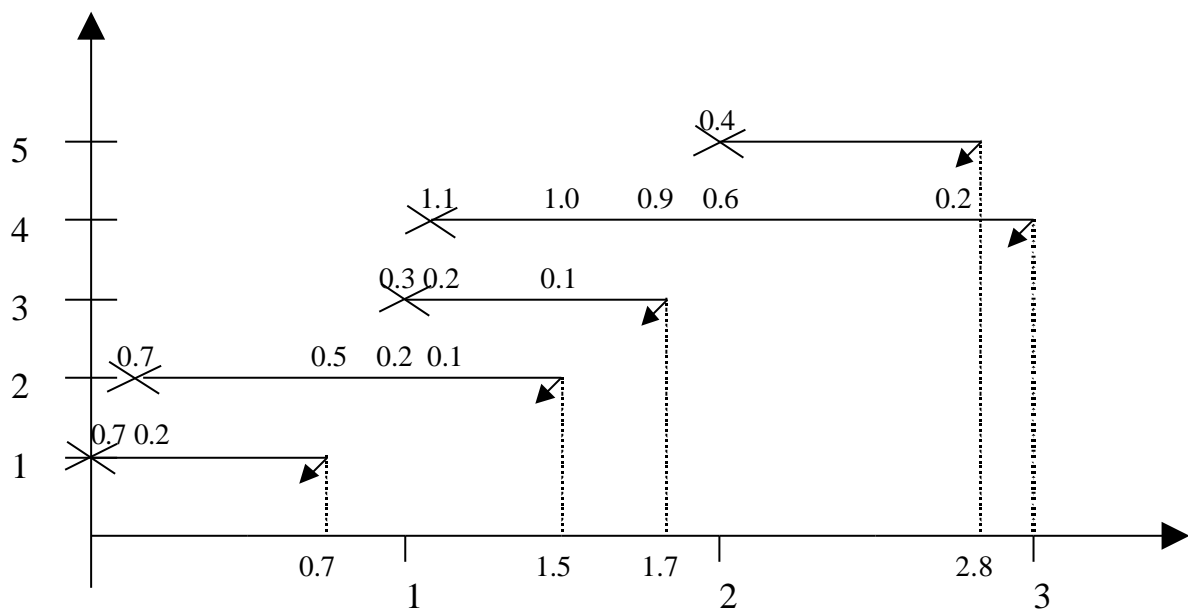
ESERCIZIO 4 (9 punti)

Sia data la seguente lista di processi (si supponga che l'istante iniziale sia 0):

Job	Tempo di Arrivo	Tempo di CPU Richiesto
1	0.0	0.5
2	0.3	0.7
3	1.0	0.3
4	1.2	1.1
5	2.0	0.4

Mostrare, utilizzando il metodo grafico, la sequenza di esecuzione dei job qualora si impieghi la politica di scheduling FIFO multiprogrammata "round robin", calcolare il tempo di *turnaround* medio e il tempo di *turnaround* pesato medio.

Soluzione:

FIFO MULTIPROGRAMMATA

Job	t_{arrivo}	t_{start}	t_{finish}	Turnaround time	Weighted Turnaround time
1	0	0	0.7	0.70	1.40
2	0.3	0.3	1.5	1.20	1.71
3	1.0	1.0	1.7	0.70	2.33
4	1.2	1.2	3	1.80	1.64
5	2.0	2.0	2.8	0.80	2.00
Media				1.04	1.82