

SOLUZIONI DELLA PROVA SCRITTA DEL CORSO DI
CALCOLATORI ELETTRONICI
 NUOVO E VECCHIO ORDINAMENTO DIDATTICO
 14 Giugno 2001

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

ESERCIZIO 1 (NO: 8 punti – VO: 7 punti)

Progettare nei seguenti modi una ALU che esegua le seguenti operazioni su due operandi A e B ad n bit:

s_0	s_1	$c=0$	$c=1$
0	0	$A+B$	$A+B+1$
0	1	B'	$-B$
1	0	$A'+B$	$A'-B'$
1	1	$A-B+1$	$A+B'+1$

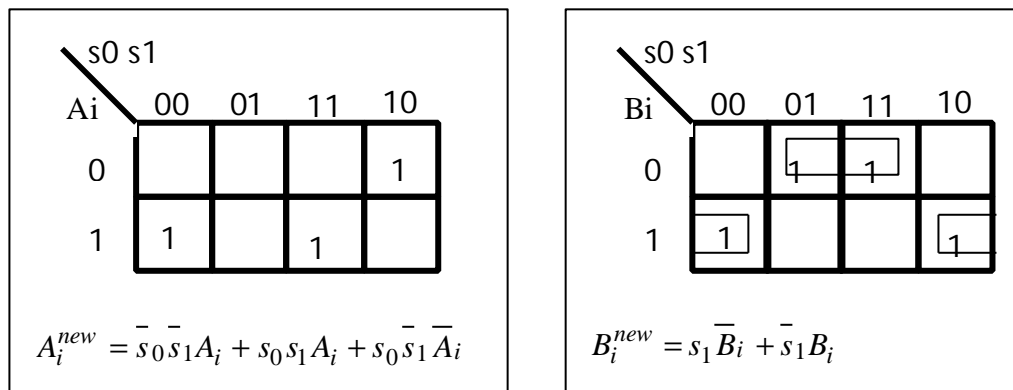
dove s_1 ed s_0 sono due variabili di controllo.

- (NO: 4 punti- VO: 3 punti) Si utilizzi un parallel adder ad n bit e una opportuna rete logica. Si tenga conto che c è il bit di riporto del parallel adder.
- (NO: 4 punti – VO: 2 punti) Si sostituisca la rete logica al passo precedente con due MUX 4-1.
- (VO: 2 punti) Si spieghi in modo chiaro e sintetico la struttura di una ALU per interi e una ALU per valori in floating point.

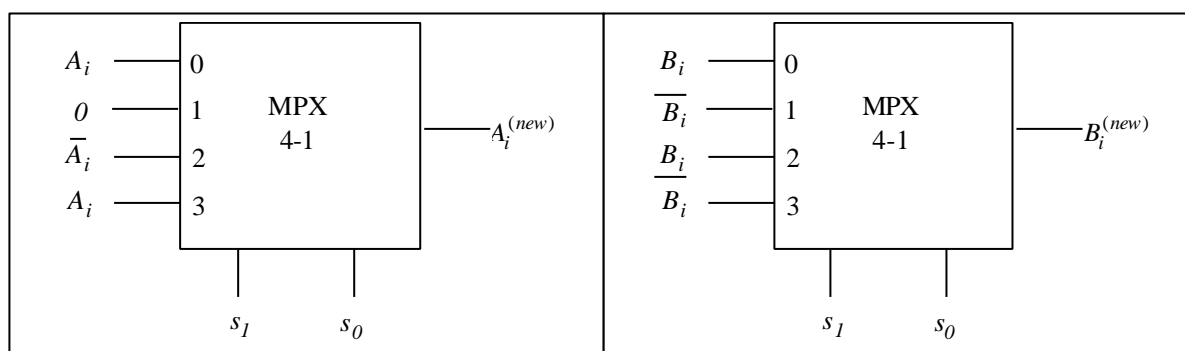
N.B.: l'apice indica la negazione.

Soluzione:

- Le mappe di Karnaugh relative alle opportune reti logiche sono:



- Utilizzando due MUX 4-1 si ha:



c) Vedi le dispense del corso.

ESERCIZIO 2 (NO: 8 punti – VO: 6 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 01001110 (il bit meno significativo è a sinistra). **Spiegando bene ogni passo del ragionamento:**

- 1) (NO: 4 punti – VO: 3) Codificare la stringa con il codice di Hamming.
- 2) (NO: 4 punti – VO: 3) 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
		0		1	0	0		1	1	1	0

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.

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

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

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

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

La stringa codificata è: 100110011110.

- 3) Per verificare la presenza di un errore nel nono bit, a causa di cui la stringa codificata è diventata 100110010110, dobbiamo ricalcolare il vettore di controllo a partire dalla sequenza ricevuta. Si ha:

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

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

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

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

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 12 bit data e precisamente nella posizione indicata dal vettore di errore tradotto in notazione decimale. Il bit sbagliato è quindi il nono (b_4), come supposto.

ESERCIZIO 3 (NO: 8 punti – VO: 7 punti)

Si consideri la sequenza di riferimenti alla memoria indicati come indirizzi di parole (il primo indirizzo è 0): 4, 17, 20, 5, 1, 43, 19, 56, 8, 4, 11, 18, 6, 9, 5, 10. Ipotizzando che la cache abbia dimensione complessiva pari a 16 parole, indicare i *cache hit* e i *cache miss* e il contenuto finale della cache nelle seguenti configurazioni:

1. cache associativa su insiemi di due blocchi con blocchi di una parola (4 punti – VO 3);
2. cache completamente associativa con blocchi di quattro parole (4 punti).

In entrambi i casi si ipotizzi una politica di sostituzione dei blocchi LRU.

Soluzione:

1. *Cache associativa su insiemi a due vie. La cache si deve considerare suddivisa in otto insiemi, ciascuno contenente due blocchi di una parola ciascuno. L'indirizzo dell'insieme in cache è pari a (indirizzo memoria) modulo (8). Cache miss/hit:*

4	17	20	5	1	43	19	56	8	4	11	18	6	9	5	10
m	m	m	m	m	m	m	m	m	h	m	m	m	m	h	m

Contenuto finale della cache:

Insieme	0	1	2	3	4	5	6	7
blocco 0	56	9	18	11	4	5	6	
blocco 1	8	1	10	19	20			

2. *Cache completamente associativa con quattro blocchi di quattro parole ciascuno. Cache miss/hit:*

4	17	20	5	1	43	19	56	8	4	11	18	6	9	5	10
m	m	m	h	m	m	m	m	m	m	h	h	h	h	h	h

Contenuto finale della cache:

blocco 0		blocco 1		blocco 2		blocco 3	
56	57	4	5	16	17	8	9
58	59	6	7	18	19	10	11

ESERCIZIO 4 (NO: 9 punti – VO: 8 punti)

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

Job	Tempo di Arrivo	Tempo di CPU Richiesto	Memoria richiesta
1	0.0	2.0	495 KB
2	0.3	0.7	195 KB
3	1.0	0.5	120 KB
4	2.0	1.3	280 KB
5	2.5	0.5	80 KB

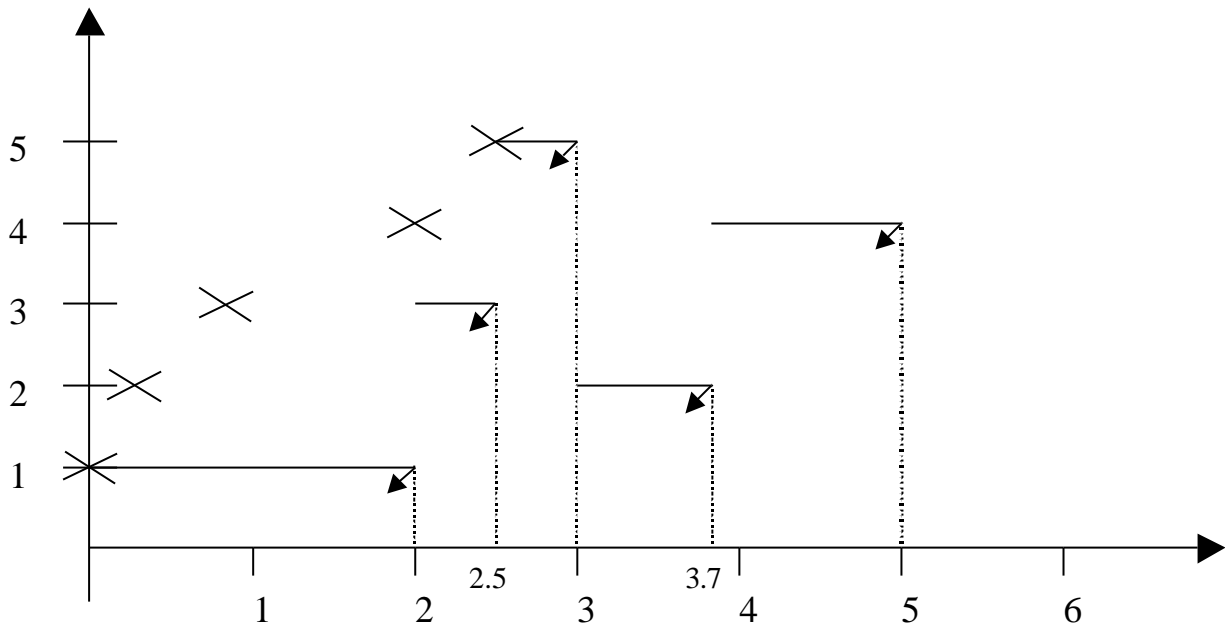
Il sistema presenta una memoria suddivisa in quattro partizioni da 300KB, 200KB, 500KB, 100KB. Nell'ipotesi che la strategia di allocazione usata sia la Best Fit:

- a) Mostrare, utilizzando il metodo grafico, la sequenza di esecuzione dei job qualora si impieghi la politica di scheduling SJF monoprogrammata (NO: 4 punti – VO: 3 punti), calcolare il tempo di *turnaround* medio e il tempo di *turnaround* pesato medio (NO: 2 punti – VO: 3 punti).

b) Mostrare in quali partizioni vengono allocati i job, giustificandone l'eventuale messa in stato di attesa (NO: 3 punti – VO: 2 punti).

Soluzione:

SJF MONOPROGRAMMATA



Job	t_{arrivo}	t_{start}	t_{finish}	Turnaround time	Weighted Turnaround time
1	0	0	2	2	1
2	0.3	3	3.7	3.4	4.86
3	1.0	2	2.5	1.5	3
4	2.0	3.7	5	3	2.31
5	2.5	2.5	3	0.5	1
Media				2.08	2.434

I Job vengono allocati nel modo seguente:

Job	Partizione	Note
1	500 KB	
2	200 KB	Allocato dopo il job 5.
3	200 KB	Allocato dopo il job 1.
4	300 KB	Allocato dopo il job 2.
5	100 KB	Allocato dopo il job 3.

ESERCIZIO 5 (VO: 5 punti)

Spiegare in modo chiaro e sintetico le principali differenze fra le architetture RISC e le architetture CISC, focalizzando in particolare il problema delle "bolle di ritardo".

Soluzione.

Si rimanda alle dispense del corso.