

**SOLUZIONI DELLA PROVA SCRITTA DEL CORSO DI
CALCOLATORI ELETTRONICI
NUOVO E VECCHIO ORDINAMENTO DIDATTICO**
17 Febbraio 2006

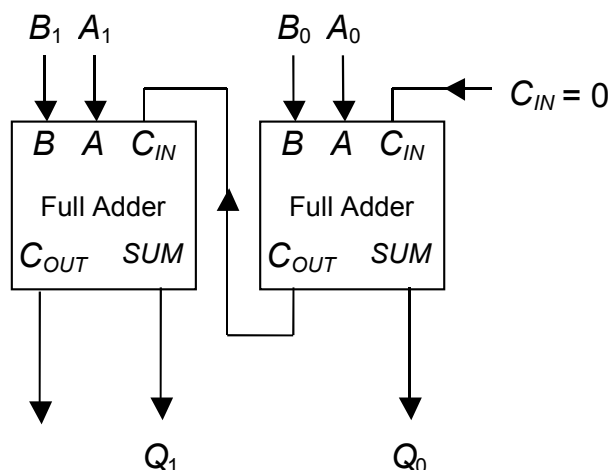
MOTIVARE IN MANIERA CHIARA LE SOLUZIONI PROPOSTE A CIASCUNO DEGLI ESERCIZI SVOLTI

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

- (1) (NO: 5 punti – VO: 4 punti) Si progetti, con il metodo delle mappe di Karnaugh, e disegni la rete logica minima che implementa un addizionatore parallelo per due addendi di due bit. Il disegno deve mostrare tutte le porte logiche ed i loro collegamenti.
- (2) (3 punti) Assumendo che una porta logica introduca un ritardo pari a T , quale è il tempo necessario ad effettuare la somma ?

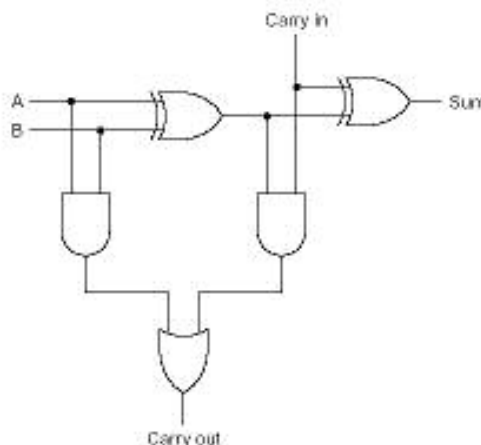
Soluzione.

Un addizionatore parallelo ("parallel adder") a due bit è costituito dalla connessione in serie di due unità dette "full adder" (v. dispense, pagg.40-41), come rappresentato in figura. Abbiamo indicato con A_1A_0 i bit del primo addendo, B_1B_0 i bit del secondo addendo (A_1 e B_1 sono i bit più significativi).



Ciascun full adder è caratterizzato dalla seguente rete logica:

A	B	Carry in	Sum	Carry out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Poiché si tratta di porte logiche a due livelli, ogni full adder produce un ritardo pari a $2T$. Conseguentemente, il ritardo complessivo del parallel adder sarà $4T$.

ESERCIZIO 2 (NO: 10 punti – VO: 8 punti)

Sia data una gerarchia di memoria costituita da memoria cache, memoria primaria e disco. Durante l'esecuzione di un processo viene rilevato che, su 1000 parole richieste, 950 sono state trovate in cache e 30 sono state trovate in memoria primaria quando non sono state trovate in cache. Istruzioni e dati del processo erano tutti memorizzati su disco. Sapendo che i tempi di accesso alla cache, alla primaria ed al disco valgono, rispettivamente, 4 nsec, 40 nsec, 2 msec, si calcoli il tempo medio di accesso alla gerarchia.

Soluzione.

Per risolvere l'esercizio è necessario calcolare l'hit ratio di cache, primaria e disco, che indichiamo nel seguito con H_c , H_p , H_d .

Nel caso di H_c , dalla definizione si ha subito $H_c = 950/1000 = 0.95$, in quanto 950 è il numero di successi in cache.

Per quanto riguarda H_p , il problema ci fornisce soltanto il numero di successi quando il dato non è presente in cache. La struttura della gerarchia però impone che i dati presenti in cache siano anche presenti in primaria, quindi il numero totale di successi in primaria è dato dalla somma dei successi in cache con quelli in primaria quando il dato non è presente in cache. In altri termini $H_p = (30 + 950)/1000 = 0.98$.

Per quanto riguarda H_d , sappiamo dal testo che istruzioni e dati sono tutti memorizzati su disco. Per cui $H_d = 1$.

Possiamo ora applicare la formula del tempo medio di accesso alla gerarchia di memoria:

$$T = H_c * T_c + (H_p - H_c) * (T_c + T_p) + (1 - H_p) * (T_c + T_p + T_d)$$

Essendo $T_c = 4 \text{ nsec}$, $T_p = 40 \text{ nsec}$, $T_d = 2 \text{ msec} = 2 * 10^6 \text{ nsec}$, si ha:

$$T = 0.95 * 4 + (0.98 - 0.95) * (4 + 40) + (1 - 0.98) * (4 + 40 + 2 * 10^6) = 3.8 + 1.32 + 40000.88 = 40006 \text{ nsec} = 40.006 \mu\text{sec}.$$

ESERCIZIO 3 (solo NO: 7 punti)

Implementare in Assembler MIPS il seguente frammento di codice C utilizzando una sola istruzione di salto:

```
for (i=0; i<N; i++)  
    x[i] += h;
```

Si ipotizzi che $\&x[0] \rightarrow$ indirizzo 30, $i \rightarrow \$8$, $h \rightarrow \$11$, $N \rightarrow \$12$. Inoltre, N sia sempre strettamente maggiore di 0. Si faccia uso di ulteriori registri se lo si ritiene necessario.

Soluzione.

Oltre ai registri indicati dal problema, utilizzeremo i seguenti:

$\$9 \leftarrow i * 4$; $\$10 \leftarrow x[i]$.

Il codice MIPS che implementa il frammento C con una sola istruzione di salto (condizionato) è il seguente:

```
        move $8, $0  
loop:   muli $9, $8, 4  
        lw  $10, 30($9)  
        add $10, $10, $11  
        sw  $10, 30($9)  
        addi $8, $8, 1  
        bne $8, $12, loop
```

ESERCIZIO 3 (solo VO: 7 punti)

Si consideri il seguente formato per la rappresentazione binaria dei numeri in virgola mobile: 24 bit, con esponente a 8 bit in eccesso **125**, mantissa frazionaria e normalizzata in segno e valore (1.M).

1. Si calcoli il minimo e il massimo numero **positivo** rappresentabili, escluso lo zero. (2 punti)
2. Si rappresentino nel formato dato i numeri 130.25 e 120.75. (2 punti)
3. Si sommino i due numeri seguendo i passi usati nell'algoritmo dei calcolatori. (3 punti)

Soluzione.

1. Minimo numero: 2^{-125} .
Massimo numero: $(2-2^{-15}) * 2^{130}$.

2.

$$130.25 = 10000010.01 = 1.000001001 * 2^7.$$

$$120.75 = 1111000.11 = 1.11100011 * 2^6.$$

$$\text{In eccesso 125: } 7 \rightarrow 111 + 1111101 = 10000100$$

$$6 \rightarrow 110 + 1111101 = 10000011$$

	S	Esponente	Mantissa
130.25	0	10000100	0000010010000000
120.75	0	10000011	1110001100000000

3. Seguendo l'algoritmo dei calcolatori, la somma si realizza coi seguenti passi:

- a) **confronto degli esponenti e allineamento.** L'esponente del primo numero è superiore a quello del secondo, per cui il secondo numero viene portato allo stesso esponente del primo. Per far ciò la mantissa scorre verso destra di una posizione, venendo così denormalizzata: $1.11100011 \rightarrow 0.111100011$
- b) **somma delle mantisse.** Ora è possibile sommare le mantisse:

$$\begin{array}{r} 1.000001001 + \\ 0.111100011 = \\ \hline 1.111101100 \end{array}$$

- c) **eventuale normalizzazione del risultato.** Non v'è alcun bisogno di normalizzare il risultato, essendo il riporto dopo il primo bit della parte intera nullo, per cui il risultato si rappresenta con lo stesso esponente e con la parte frazionaria della mantissa:

	S	Esponente	Mantissa
251.00	0	10000100	1111011000000000

ESERCIZIO 4 (NO: 8 punti – VO: 6 punti)

Si consideri un disco rigido con le seguenti caratteristiche: velocità di rotazione = 5400 giri/min, tempo medio di posizionamento = 5 ms, 34 settori per traccia di 512 byte ciascuno.

- a) (NO: 4 punti – VO: 3 punti) Calcolare il tempo medio di trasferimento di un file da 8 kbyte considerando il file posizionato su settori collocati in tracce non contigue e la testina all'istante iniziale sul primo settore del file.
- b) (NO: 4 punti – VO: 3 punti) Supporre che il file debba essere trasferito in memoria principale, attraverso un modulo DMA. Ipotizzando una linea dati di 8 bit, una velocità di clock di 2 GHz, e che il buffer del DMA sia stato caricato con tutto il file, qual è il tempo minimo necessario per il trasferimento?

Soluzione

a)

$$T_{rot} = 60/5400 \text{ sec} = 11.11 \text{ msec.}$$

$$T_{lat} = T_{rot}/2 = 5.555 \text{ msec.}$$

$$T_{ls} = T_{rot} / 34 = 0.327 \text{ msec. (tempo di lettura di un settore)}$$

$$T_{pos} = 5 \text{ msec.}$$

$$\text{Numero di settori richiesti dal file } N = 8\text{kbyte} / 512\text{byte} = 16 \text{ settori.}$$

$$T_{lett} = (N-1) * (T_{lat} + T_{pos} + T_{ls}) + T_{ls} = 15 * 10.882 + 0.327 = 163.557 \text{ msec}$$

b) In tal caso viene trasferito un byte ogni 0.5 nsec (block transfer). Quindi il tempo complessivo è $8 * 1024 * 0.5 \text{ nsec} = 4096 \text{ nsec} = 4.1 \mu\text{sec}$ (circa).

ESERCIZIO 5 (solo VO: 5 punti)

Spiegare in modo chiaro e sintetico la classificazione di Flynn delle architetture parallele.

Soluzione.

Vedi dispense del corso.