

PROVA SCRITTA DEL MODULO DI
CALCOLATORI ELETTRONICI
NUOVO E VECCHIO ORDINAMENTO DIDATTICO (5-7 CFU)

17 settembre 2015

NOME:

COGNOME:

MATRICOLA:

CFU:

MOTIVARE LA SOLUZIONE PROPOSTA A CIASCUNO DEGLI ESERCIZI SVOLTI

ESERCIZIO 1 (8 punti)

1. (4 punti) Progettare un Full Adder. Mostrare la tabella di verità e le porte logiche che implementano tale rete combinatoria.
2. (4 punti) Disegnare lo schema di un Parallel Adder con due addendi di quattro bit, precisando il ruolo di ogni Full Adder componente. Indicare cos'è e come viene calcolato il tempo di ritardo introdotto da un Parallel Adder. Assumere che 'd' sia il ritardo introdotto da un singolo modulo Full Adder.

ESERCIZIO 2 (8 punti)

1. (4 punti) Sia dato un disco con le seguenti caratteristiche: velocità di rotazione pari a 6000 giri al minuto, tempo necessario alla testina per spostarsi da una traccia alla successiva uguale a 0.5 ms, settori da 1 KB, 200 settori per traccia. Calcolare il tempo medio di lettura di un file da 10 KB sapendo che la testina si trova inizialmente in un punto qualunque del disco e che la distanza media tra due settori successivi del file è pari a 2 tracce.
2. (4 punti) Utilizzando il tempo medio di accesso al disco calcolato al punto precedente, si calcoli l'hit ratio di cache minimo (H_c), relativo ad una gerarchia a 3 livelli (cache, memoria primaria, disco), affinché il tempo medio di accesso alla gerarchia sia 30 ms. Siano dati i seguenti altri dati: $H_p = 0.8$; $T_p = 20$ ms; $T_c = 10$ ms, dove H_p è l'hit ratio della memoria primaria, T_p il tempo medio di accesso alla memoria primaria e T_c il tempo medio di accesso alla memoria cache.

ESERCIZIO 3 (9 punti)

Implementare una procedura Assembly MIPS chiamata "resti" che, dati l'indirizzo iniziale di due vettori v (in \$4) e w (in \$5), di dimensione N in \$6, generi un nuovo vettore z a partire da un indirizzo presente in \$7, tale che, per $i=0, \dots, N-1$, $z[i]=v[i] \bmod w[i]$, dove "mod" indica il resto della divisione tra $v[i]$ e $w[i]$. Si faccia uso di una funzione già implementata, `mod`, che prevede in \$4 il dividendo ed in \$5 il divisore, mentre il resto viene depositato in \$6.

ESERCIZIO 4 (8 punti)

Si consideri un calcolatore in cui la CPU esegue 10^5 istruzioni/s. L'esecuzione di una istruzione richiede 5 cicli di clock, 3 dei quali tengono occupato il bus di sistema. Si ipotizzi che il 75% dell'Instruction Rate sia usato dalla CPU per eseguire programmi che non contengono trasferimenti di I/O. L'ampiezza della linea dati del bus è pari a 32 bit.

Si consideri il caso in cui il trasferimento dei dati avvenga mediante IO da programma, con le seguenti 4 istruzioni:

- 1) `LOAD parola` dalla periferica al registro CPU
 - 2) `STORE parola` da registro CPU a memoria
 - 3) generazione indirizzo di memoria successivo
 - 4) conteggio dati da trasferire.
- a) (3 punti) Calcolare la massima frequenza di trasferimento dati ottenibile (espressa in kB/s) fra una periferica collegata al bus di sistema e la memoria principale mediante I/O da programma, sapendo che una parola è pari a 32 bit.
 - b) (3 punti) Calcolare la massima frequenza di trasferimento dati ottenibile (espressa in kB/s) nel caso in cui si usi la modalità "transparent" DMA. Si ipotizzi che una operazione di lettura/scrittura della memoria richieda un ciclo di clock e che una parola sia pari a 32 bit.
 - c) (2 punti) Si descrivano, in modo chiaro e sintetico, le differenze tra il sistema di arbitraggio centralizzato e quello distribuito, facendo particolare riferimento ai sistemi di gestione delle periferiche "daisy chain" e "a richieste indipendenti".

ESERCIZIO 1

Soluzione

Si vedano le dispense del corso, capitolo 7, pp. 45-46.

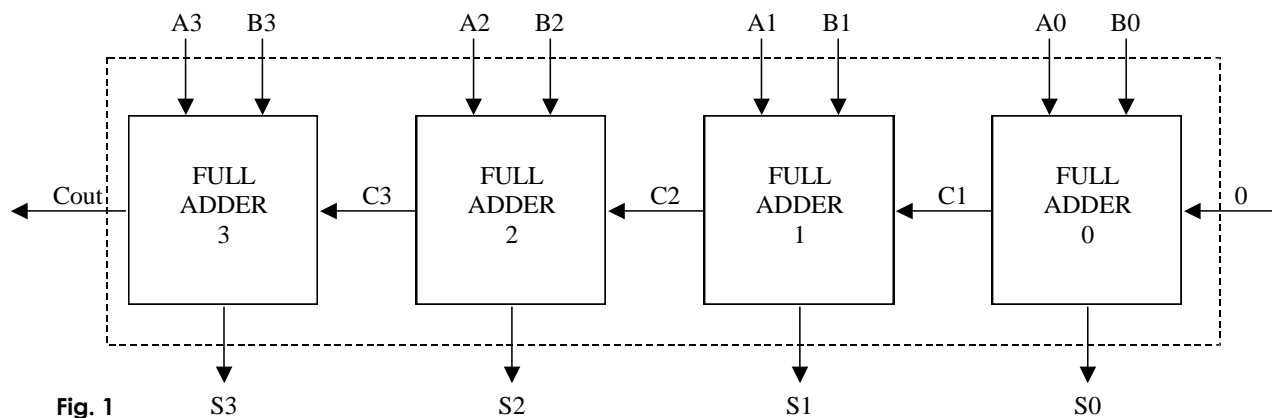


Fig. 1

Fig.1 mostra lo schema di un Parallel Adder con due addendi di quattro bit, indicati con A e B rispettivamente (A₀ e B₀ sono i bit meno significativi). C₀ è il riporto in ingresso, Cout quello in uscita, C₁...C₃ i riporti intermedi. Se d è il tempo di ritardo del FA singolo, il tempo di ritardo complessivo è dato da 4*d. Ricordiamo che il tempo di ritardo è il tempo richiesto al Parallel Adder per presentare le quattro uscite S₀...S₃ (il risultato della somma A+B). Poiché ogni bit somma S_i dipende dal ritardo degli stadi precedenti, il ritardo complessivo è dato dalla somma dei ritardi dei singoli FA.

ESERCIZIO 2**Soluzione**

- $TROT = 60 / 6000 = 10 \text{ ms}$
 $TLAT = TROT / 2 = 5 \text{ ms}$ (tempo di latenza)
 $Tlett = TROT / 200 = 50 \mu\text{s}$ (tempo di lettura di un settore)
 $Tpos = 0.5 \times 2 = 1 \text{ ms}$
 Numero di settori richiesti per leggere 10 KB: $10\text{KB}/1\text{KB} = 10$
 Tempo di lettura del file da 10 KB:
 $= 10 * (TLAT + TPOS + Tlett) = 10 * (5 \text{ ms} + 1 \text{ ms} + 50 \mu\text{s}) \cong 60 \text{ msec}$
- Per calcolare H_c basta esplicitarlo dalla formula:

$$\bar{T} = H_c T_C + (H_p - H_c)(T_p + T_C) + (1 - H_p)(T_D + T_p + T_C)$$

$$H_c = (T_C + T_p + T_D - H_p T_D - \bar{T}) / T_p = 0.6$$

ESERCIZIO 3**Soluzione.**

Utilizziamo i seguenti registri:

$\$8 \leftarrow i; \$9 \leftarrow 4i; \$1 \leftarrow \&v[i], \&w[i], \&z[i]$

```

resti:      addi $29, $29, -28
            sw $8, 0($29)
            sw $9, 4($29)
            sw $1, 8($29)
            sw $20, 12($29)
            sw $21, 16($29)
            sw $22, 20($29)
            sw $31, 24($29)
            move $20, $4
            move $21, $5
            move $22, $6
            move $8, $0
for:        beq $8, $22, exit
            muli $9, $8, 4
            add $1, $9, $20
            lw $4, 0($1)
            add $1, $9, $21
            lw $5, 0($1)
            jal mod
            add $1, $9, $7
            sw $6, 0($1)
            addi $8, $8, 1
            j for
exit:       lw $8, 0($29)
            lw $9, 4($29)
            lw $1, 8($29)
            lw $20, 12($29)
            lw $21, 16($29)
            lw $22, 20($29)
            lw $31, 24($29)
            addi $29, $29, 28
            jr $31

```

ESERCIZIO 4

Soluzione

- a) Nel caso di trasferimento mediante I/O da programma, per trasferire una parola occorrono 4 istruzioni. La CPU è impegnata per il 75% del tempo a eseguire istruzioni che non coinvolgono l'I/O, dunque può usare solo il 25% del tempo per eseguire istruzioni di trasferimento dati con periferiche. In termini di istr./sec questo tempo è pari a $0.25 \times 10^5 \text{ istr./s} = 2.5 \times 10^4 \text{ istr./s}$. Dal momento che per trasferire una parola servono quattro istruzioni, la velocità di trasferimento è pari a:

$2.5 \times 10^4 \text{ istr./s} / (4 \text{ istr./parola}) = 6250 \text{ parole/s}$. La dimensione di una parola è pari a 32 bit (4 byte), da cui si ricava la velocità di trasferimento di **24.41 kB/s**,

- b) Nel caso di 'transparent DMA' posso trasferire i dati tutte le volte che il bus di sistema è libero. Nel caso in esame questo tempo è pari alla somma del 25% del tempo lasciato libero dall'esecuzione di istruzioni che non coinvolgono I/O, più i due cicli/istruzione in cui il bus è libero. Pertanto durante il 75% del tempo posso trasferire una parola/istr.:

$$0.75 \times 2 \text{ parole/istr} \times 10^5 \text{ istr./s} = 1.5 \times 10^5 \text{ parole/s}$$

Nel restante 25% del tempo posso trasferire 5 parole/istr.:

$$0.25 \times 5 \text{ parole/istr.} \times 10^5 \text{ istr./s} = 1.25 \times 10^5 \text{ parole/s}$$

In **totale**, nel caso di trasferimento con DMA la velocità totale di trasferimento è pari a: **$(1.5 + 1.25) \times 10^5 \text{ parole/s} = 2.75 \times 10^5 \text{ parole/s} = 1074.22 \text{ kB/s}$**

- c) si vedano le dispense del corso.