

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

13 Luglio 2004

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

ESERCIZIO 1 (9 punti)

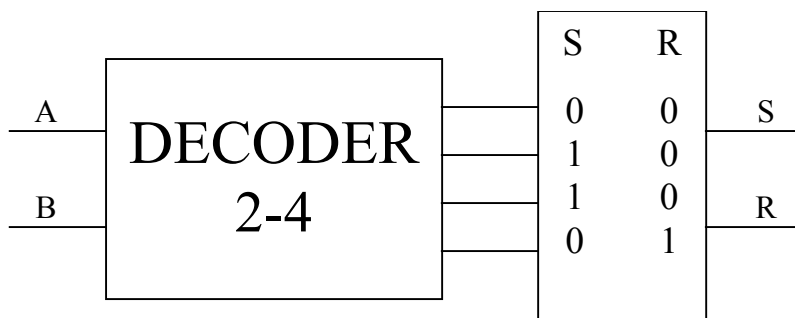
Si vuole realizzare una rete logica che dati due ingressi A, B, produca le due uscite S ed R definite dalle due seguenti funzioni booleane: $S = AB' + A'B$, $R = AB$. L'apice indica la negazione.

- a) (2 punti) spiegare chiaramente ed in maniera motivata quale è la funzione realizzata da tale rete logica;
- b) (3 punti) realizzare tale rete utilizzando un "decoder" ed una memoria ROM;
- c) (2 punti) sintetizzare tale rete con il minor numero di porte logiche (disegnare il circuito);
- d) (2 punti) discutere brevemente, ma chiaramente, vantaggi e svantaggi delle due precedenti realizzazioni.

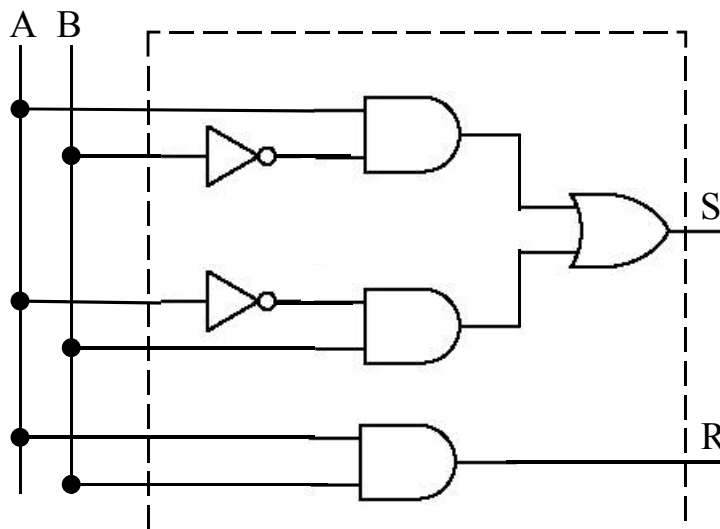
Soluzione.

a) Si tratta ovviamente di un half adder.

b)



c)



d) Vedi dispense del corso, cap.3, pagg. 6-9.

ESERCIZIO 2 (NO: 7 punti – VO: 8 punti)

Si consideri una memoria primaria costituita da 8 parole indirizzabili e da una cache formata da quattro parole.

- 1) (2 punti) Spiegare, precisando il significato e la funzione dei diversi campi, come vengono interpretati gli indirizzi logici per recuperare l'informazione contenuta nella cache nel caso venga usata la modalità di indirizzamento:
 - (a) Diretto, con blocchi di due parole
 - (b) "associativo su insiemi", con insiemi di due blocchi di una parola ciascuno.
- 2) (NO: 5 punti – VO: 4 punti) Ipotizzando la cache vuota all'istante iniziale, indicare nei casi (a-b) lo stato finale della cache e il numero di hit nel caso vengano inoltrate le seguenti chiamate (indirizzi espressi in decimale, primo indirizzo 0): 4, 6, 6, 5, 1, 3, 7, 6, 3, 6. Nel caso (b) si ipotizzi una politica di sostituzione dei blocchi FIFO.
- 3) (solo VO: 2 punti) Spiegare in modo chiaro e sintetico quali sono i registri della CPU coinvolti nella lettura/scrittura di una parola dalla/sulla memoria.

Soluzione.

1)

Memoria indirizzabile: 8 parole = 2^3 parole → 3 bit di indirizzamento

(a) <TAG 1 bit> <Cache Index 1 bit> <Offset 1 bit>

(b) <TAG 2 bit> <Cache Index 1 bit>

2) Nel seguito, è disegnata la cache di quattro parole, e il suo stato finale nei casi a-b. Con linee tratteggiate indichiamo la separazione tra un blocco e il successivo, con linee continue la separazione tra due insiemi.

(a)

0
1
6
7

(b)

4
6
3
7

Tale stato finale è dovuto alle seguenti dinamiche. La prima riga in tabella riporta la parola chiamata, l'ultima il verificarsi di un hit.

Metodo Diretto

4	6	6	5	1	3	7	6	3	6
4	4	4	4	0	0	0	0	0	0
5	5	5	5	1	1	1	1	1	1
	6	6	6	6	2	6	6	2	6
	7	7	7	7	3	7	7	3	7
			h	h			h		

Metodo associativo su insiemi

4	6	6	5	1	3	7	6	3	6
4	4	4	4	4	4	4	4	4	4
	6	6	6	6	6	6	6	6	6
			5	5	3	3	3	3	3
				1	1	7	7	7	7
			h				h	h	h

Metodo diretto: 3/10 hit

Metodo associativo su insiemi: 4/10 hit

- 3) I registri interessati sono il MAR e l'MBR: Memory Address Register che conserva l'indirizzo della parola da prelevare/traferire, e il Memory Buffer Register che conserva il dato prelevato/trasferito. Vedi anche dispense del corso.

ESERCIZIO 3 (solo NO: 10 punti)

a) (8 punti) Il seguente codice Assembler MIPS dovrebbe implementare una funzione che calcola la media m dei valori di un vettore v di N elementi, secondo la formula:

$$m = \frac{1}{N} \sum_{i=1}^N v_i$$

I parametri di ingresso sono $\&v[0] \rightarrow \$4$, $N \rightarrow \$5$, $m \rightarrow \$6$. Per la divisione x/y la funzione fa uso di un'altra funzione `div` i cui parametri di ingresso $x \rightarrow \$4$, $y \rightarrow \$5$, $x/y \rightarrow \$6$.

Trovare eventuali errori presenti nel codice dato.

b) (2 punti) Spiegare in modo chiaro e sintetico quali sono i registri della CPU coinvolti nella fase di fetch del ciclo di esecuzione delle istruzioni.

```
media:      addi $29, $29, -12
            sw $8, 0($29)
            sw $9, 4($29)
            move $6, $0
            move $8, $0
for:        bne $8, $5, exit
            lw $9, $0($4)
            add $6, $6, $9
            addi $8, $8, 1
            addi $4, $4, 1
exit:       jr div
            lw $8, 0($29)
            lw $9, 4($29)
            lw $31, 8($29)
            addi $29, $29, 20
            j $31
```

Soluzione.

```
media:      addi $29, $29, -12
            sw $8, 0($29)
            sw $9, 4($29)
            sw $31, 8($29)      → istruzione omessa nell'originale
            move $6, $0
            move $8, $0
for:        beq $8, $5, exit    → sostituito beq a beq
            lw $9, 0($4)       → sostituito 0 a $0
            add $6, $6, $9
            addi $8, $8, 1
            addi $4, $4, 4     → sostituito 4 a 1
            j for              → istruzione omessa nell'originale
exit:       move $4, $6        → istruzione omessa nell'originale
            jal div            → sostituito jal a jr
            lw $8, 0($29)
            lw $9, 4($29)
            lw $31, 8($29)
            addi $29, $29, 12   → sostituito 12 a 20
            jr $31             → sostituito jr a j
```

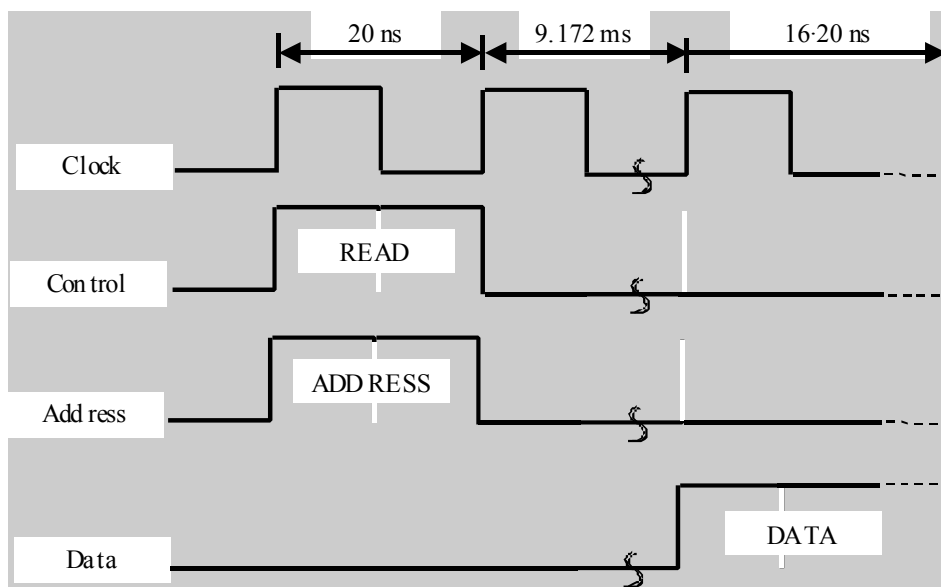
b) I registri interessati sono il MAR e l'MBR: Memory Address Register (MAR) che conserva l'indirizzo della istruzione da prelevare passata dal Program Counter (PC), e il Memory Buffer Register che conserva l'istruzione prelevata che verrà copiata nell'Instruction Register (IR). Vedi anche dispense del corso, cap.9, pag.7 e segg.

ESERCIZIO 3 (solo VO: 6 punti)

- (a) (3 punti) Un disco rigido presenta le seguenti caratteristiche: velocità di rotazione 7200 giri/min; tempo medio di posizionamento 5 msec; capacità di una traccia 100 kbyte. Calcolare il tempo medio di lettura di un blocco di 64 byte.
- (b) (3 punti) I trasferimenti di dati dal disco descritto al punto precedente alla memoria avvengono su un bus sincrono, il cui clock ha frequenza pari a 50 MHz. Sapendo che ciascun trasferimento sul bus ha una durata pari a un ciclo di clock e che l'ampiezza della linea dati del bus è di 32 bit, calcolare il tempo totale necessario per il trasferimento di un blocco di 64 byte dal disco alla memoria principale. Nello svolgimento della soluzione, mostrare il diagramma temporale dei trasferimenti sul bus. [Nota: il disco trasferisce tutto il blocco richiesto in un *buffer* che è collegato al bus di sistema].

Soluzione

- 1) Tempo di rotazione $t_{rot} = 60/7200 = 8.33 \text{ ms}$
Tempo di latenza $t_{lat} = 1/2 t_{rot} = 4.167 \text{ ms}$
Tempo di trasferimento di un blocco da 64 byte $t_{trasf} = 64 \cdot 60 / (100k \cdot 7200) = 5.21 \text{ } \mu\text{s}$
Tempo di lettura totale $t_{lett} = t_{pos} + t_{lat} + t_{trasf} = 9.172 \text{ ms}$
(NOTA il tempo di trasferimento è praticamente irrilevante).
- 2) In un bus sincrono il trasferimento avviene in tre passi: invio del segnale di READ sulla linea di controllo e contemporaneamente l'indirizzo di memoria sulla linea indirizzi; tempo di "latenza" dovuto al ritardo con cui la periferica rende disponibili i dati nel buffer di uscita (nel caso del disco questo tempo è pari a 9.167 ms); trasferimento dei dati dal buffer alla memoria. In questo caso dovendo trasferire 64 byte su un bus con ampiezza dati di 32 bit, saranno necessari 16 trasferimenti consecutivi.
Tempo totale di trasferimento: $T(\text{READ \& ADDRESS}) + T_{disco} + T_{trasf} = 20\text{ns} + 9.172\text{ms} + 16 \cdot 20\text{ns} = 9.17234 \text{ ms}$. (NOTA: il "collo di bottiglia" è il disco).



ESERCIZIO 4 (NO: 7 punti – VO: 5 punti)

La memoria di un calcolatore è gestita con una tecnica di 'paginazione su richiesta'. Si consideri la seguente richiesta di pagine: 0, 7, 2, 7, 5, 8, 9, 2, 4, 5, 2, 5. Mostrare il "page trace" e calcolare il numero di "page faults" per una memoria di dimensione pari a quattro pagine, nei seguenti due casi:

- 1) (NO: 3 punti – VO: 2 punti) strategia di rimpiazzamento delle pagine FIFO.
- 2) (NO: 4 punti – VO: 3 punti) strategia di rimpiazzamento delle pagine LRU.

Soluzione

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
Richieste	0	7	2	7	5	8	9	2	4	5	2	5
Pagine	0	7	2	2	5	8	9	9	4	4	2	5
		0	7	7	2	5	8	8	9	9	4	2
			0	0	7	2	5	5	8	8	9	4
					0	7	2	2	5	5	8	9
Hit				x				x		x		

Page faults: 9 - Hit ratio = $3/12 = 1/4$

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
Richieste	0	7	2	7	5	8	9	2	4	5	2	5
Pagine	0	7	2	7	5	8	9	2	4	5	2	5
		0	7	2	7	5	8	9	2	4	5	2
			0	0	2	7	5	8	9	2	4	4
					0	2	7	5	8	9	9	9
Hit				x							x	x

Page faults: 9 - Hit ratio = $3/12 = 1/4$

ESERCIZIO 5 (solo VO: 5 punti)

Si consideri un calcolatore multiprocessore con p processori. Il tempo di esecuzione di una certa applicazione M su tale calcolatore sia pari a $40/p$, e il tempo necessario per le comunicazioni fra processori sia pari a $10 \cdot \sqrt{p}$ ($p > 1$).

- (a) (1 punto) Calcolare il tempo necessario per eseguire l'applicazione in un'architettura monoprocessore.
- (b) (3 punti) Calcolare il numero di processori che massimizza lo "speedup".
- (c) (1 punto) Calcolare il valore dello "speedup" e dell'"efficienza" relativi al numero di processori per cui lo speedup è massimo.

Soluzione

- 2) Il tempo necessario per eseguire un programma su un'architettura parallela è espresso come la somma di due termini: il tempo necessario per effettuare le elaborazioni locali che vengono eseguite in "parallelo" su p processori e il tempo necessario per lo scambio dei dati e la sincronizzazione fra i diversi processori.

Il tempo $T(p)$ di esecuzione dell'applicazione M in un'architettura multiprocessore è pari a $40/p + 10 \cdot \sqrt{p}$.

Per $p = 1 \rightarrow T(1) = 40$, tempo di esecuzione di M in un'architettura monoprocessore.

- 3) Lo "Speedup" $S(p)$ è definito come il rapporto $T(1)/T(p)$. Per massimizzare $S(p)$ si deve calcolare il numero di processori che minimizza il valore di $T(p)$. Ad esempio derivando l'espressione di $T(p)$ e uguagliando a zero si ottiene:

$$\frac{dT(p)}{dp} = -\frac{40}{p^2} + \frac{10}{2\sqrt{p}} = 0 \Rightarrow -8 + p \cdot \sqrt{p} = 0 \Rightarrow p^3 = 64 \Rightarrow p = 4$$

Pertanto lo speedup è massimo per $p = 4$

- 4) Per $p = 4$ lo speedup $S = 1.33$ e l'efficienza $E = S(p)/p = 0.33$. Si ha una riduzione dello 0,03%.