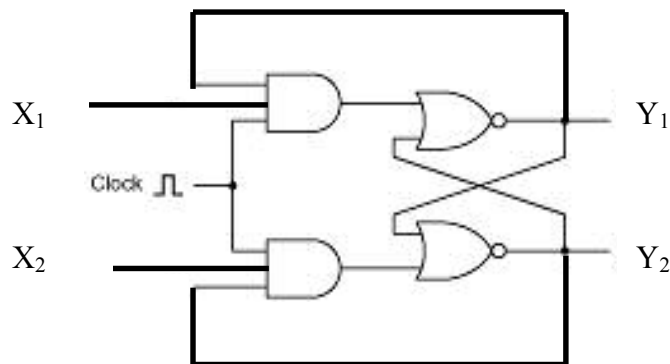


**SOLUZIONI DELLA PROVA SCRITTA DEL CORSO DI  
CALCOLATORI ELETTRONICI  
NUOVO ORDINAMENTO DIDATTICO**  
5 Luglio 2005

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

**ESERCIZIO 1 (8 punti)**

- 1) (2 punti) Indicare a quale categoria di reti logiche appartiene il circuito in figura (combinatorie o sequenziali) e spiegare il perché.
- 2) (4 punti) Analizzare la rete logica in figura, scrivendo l'espressione delle uscite in funzione degli ingressi e la corrispondente tabella di verità. Spiegare se si tratta di una rete logica conosciuta, e indicare nel caso quale sia, precisando il significato degli ingressi e delle uscite.  
(Suggerimento: si consideri che vale la relazione  $Y_2 = \overline{Y_1}$ ).
- 3) (2 punti) Se si connettono tra loro gli ingressi della rete logica in figura, cosa si ottiene?



Soluzione.

- 1) Si tratta di una rete logica sequenziale, in quanto le uscite sono retroazionate in ingresso.
- 2) Dall'analisi del circuito, tenendo conto del suggerimento, otteniamo la seguente espressione:

$Y_1(t + \tau) = Y_1(t)\overline{X_1} + \overline{Y_1(t)}X_2$ , che si traduce nella seguente tabella di verità:

$X_1$	$X_2$	$Y_1(t)$	$Y_1(t+\tau)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

Si tratta evidentemente della tabella di transizione del flip flop JK, in cui  $X_1 = K$ ,  $X_2 = J$ . L'uscita  $Y_1$  corrisponde invece allo stato del FF, in particolare al valore Q.

L'esercizio si risolve anche in funzione di  $Y_2$ , corrispondente a Q'.

- 3) Dato che la rete è un FF-JK, connettendo tra loro gli ingressi si ottiene un FF-T, con  $T = J = K$ .

## ESERCIZIO 2 (8 punti)

Si consideri una gerarchia di memoria cache-primaria, con una memoria cache di 32 parole ed una memoria primaria di 256 parole. Si considerino i seguenti metodi di indirizzamento:

- Diretto, con blocchi di due parole.
  - Associativo su insiemi, con insiemi di due blocchi e blocchi di due parole.
- 1) (2 punti) Indicare, specificando il significato e la funzione dei diversi campi, come viene recuperata l'informazione nella cache a partire dall'indirizzo della parola in memoria primaria nei due casi.
  - 2) (6 punti) Si ipotizzi che, a cache inizialmente vuota, il sistema richieda l'accesso alle parole da 0 a 15 e da 32 a 47 in questo ordine. Indicare lo stato finale della cace e calcolare l'hit ratio relativo a tali accessi per ciascuno dei due metodi di indirizzamento.

### Soluzione.

1)

Indirizzamento: 8 bit

Diretto:

< Tag 3 bit > < Cache Index 4 bit > < Offset 1 bit >

Associativo su insiemi:

< Tag 4 bit > < Cache Index 3 bit > < Offset 1 bit >

2)

Sulla base dei valori trovati al punto precedente, si calcola facilmente che lo stato della cache dopo la sequenza 0-15, 32-47 è data nel primo caso da:

Blocchi cache →	0	1	2	3
0	32	34	36	38
1	33	35	37	39
Blocchi cache →	4	5	6	7
0	40	42	44	46
1	41	43	45	47
Blocchi cache →	8	9	10	11
0				
1				
Blocchi cache →	12	13	14	15
0				
1				

Nel secondo caso si ha invece:

Insiemi cache →	0	1	2	3
0	0	2	4	6
1	1	3	5	7
0	32	34	36	38
1	33	35	37	39
Insiemi cache →	5	6	7	8
0	8	10	12	14
1	9	11	13	15
0	40	42	44	46
1	41	43	45	47

In entrambi i casi, l'hit ratio è dato da:

- 8 hit per gli accessi da 0 a 15
- 8 hit per gli accessi da 32 a 47.
- 

Essendo il numero complessivo di accessi pari a 32, l'hit ratio vale  $16/32 = 0.5$ .

### ESERCIZIO 3 (9 punti)

Implementare in Assembler MIPS una funzione tale che, dati gli indirizzi iniziali di due vettori  $v$  (in \$4) e  $w$  (in \$5) e la relativa dimensione  $N$  (in \$6), scambi i valori  $v[i]$  con  $w[i]$  se  $v[i] < w[i]$ . Si assuma, per lo scambio, di disporre di una funzione `scambia` che riceva in ingresso gli indirizzi iniziali dei due vettori (in \$4 e \$5) e l'indice  $i$  (in \$6).

La funzione richiesta potrebbe essere la traduzione del seguente codice C:

```
void elabora_vettori(int *v, int *w, int N)
{
    int i;
    for(i=0; i<N; i++)
        if(v[i]<w[i])
            scambia(v,w,i);
}
```

All'uscita della funzione, i valori iniziali di \$4, \$5, \$6, devono risultare inalterati.

#### Soluzione.

1)

```
$8 ← copia di N; $9 ← i*4; $10 ← v+i*4; $11 ← w+i*4;
$12 ← v[i]; $13 ← w[i]; $14 ← (v[i]<w[i])
```

```
elabora_vettori:  addi $29, $29, -32
                  sw $8, 0($29)
                  ...
                  sw $14, 24($29)
                  sw $31, 28($29)
                  move $8, $6
                  move $6, $0
for:              beq $6, $8, exit
                  muli $9, $6, 4
                  add $10, $4, $9
                  add $11, $5, $9
                  lw $12, 0($10)
                  lw $13, 0($11)
                  slt $14, $12, $13
                  beq $14, $0, aggiorna
                  jal scambia
aggiorna:         addi $6, $6, 1
                  j for
exit:             move $6, $8
                  lw $8, 0($29)
                  ...
                  lw $14, 24($29)
                  lw $31, 28($29)
                  addi $29, $29, 32
                  jr $31
```

**ESERCIZIO 4 (8 punti)**

Si consideri un disco con le seguenti caratteristiche: 5400 giri/minuto, 100 settori/traccia, 8 KB/settore, 30 tracce, tempo di posizionamento da una traccia a quella adiacente pari a 1 ms. Calcolare il tempo di lettura di un file da 128 KB, considerando che il file è stato registrato nel seguente modo:

- settori del file ad indirizzo pari contigui a partire dal primo settore della prima traccia;
- settori del file ad indirizzo dispari contigui a partire dal secondo settore dell'ultima traccia;
- testina del disco in un settore qualsiasi della prima traccia all'istante iniziale.

**Soluzione.**

$$T_L = \frac{1}{2} T_{\text{rot}} = \frac{1}{2} * 11,1 \text{ ms} = 5,6 \text{ ms.}$$

$$T_{\text{lett}} = T_{\text{rot}} / 100 = 0,111 \text{ ms.}$$

$$T_p = 1 \text{ ms}$$

$$N_{\text{sett}} = 128 \text{ KB} / 8 \text{ KB/settore} = 16 \text{ settori}$$

I tempi impiegati per la lettura del file sono i seguenti:

- 1) Latenza + lettura, primo settore
- 2) 29 spostamenti lungo le tracce + lettura per i restanti quindici settori

In altri termini:

$$T = 16 * T_{\text{lett}} + T_L + 15 * 29 * T_p = (16 * 0,111 + 5,6 + 15 * 29 * 1) \text{ ms} = 1,776 + 5,6 + 435 = 442,376 \text{ ms.}$$