

PRIMA PROVA INTERMEDIA DEL CORSO DI
CALCOLATORI ELETTRONICI
NUOVO ORDINAMENTO DIDATTICO – A.A. 2015/16
22 aprile 2016

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

NOME: _____ COGNOME: _____ MATRICOLA: _____

ESERCIZIO 1 (8 punti)

Progettare un riconoscitore della sequenza 0010. Si usino FF-T per il modulo di memorizzazione dello stato. Semplificare le espressioni delle reti di transizione dello stato con le mappe di Karnaugh. Si calcoli anche la rete logica di transizione dell'uscita.

ESERCIZIO 2 (8 punti)

Si consideri una gerarchia di memoria cache-primaria. La primaria è costituita da 2K parole, con blocchi di 16 parole. La cache sia costituita da 128 parole. L'indirizzamento utilizzato per l'associazione blocchi-linee è quello diretto.

1. (2 punti) Spiegare come vengono interpretati gli indirizzi di memoria primaria per recuperare l'informazione contenuta nella cache.
2. (5 punti) Indicare, attraverso i valori di block frame, quali blocchi di memoria primaria possono essere associati alla linea di cache indirizzata con un valore di cache index uguale a 4.
3. (1 punto) Calcolare il tempo medio di accesso alla gerarchia considerando i seguenti valori: $H_c=0.95$, $T_c=4\text{ ns}$, $T_p=40\text{ ns}$.

ESERCIZIO 3 (7 punti)

E' data un'unità a disco caratterizzata dai seguenti parametri: 6000 giri/min, 128 tracce, 64 settori per traccia, 4 KB/settore, tempo di posizionamento da una traccia a quella adiacente pari a 1 ms.

1. (2 punti) Volendo scrivere un file di 1 MB su disco su settori e tracce contigue (in modalità non frammentata), quante tracce e quanti settori occorrono?
2. (5 punti) Si calcoli il tempo di scrittura nella modalità di cui sopra, nel caso la testina si trovi inizialmente nell'ultima traccia ed il primo settore utile si trovi in un punto qualsiasi della prima.

ESERCIZIO 4 (6 punti)

Una parola di 8 bit va trasferita dalla CPU alla memoria con il codice di Hamming.

1. (1 punto) Calcolare il numero di bit di controllo per la codifica della parola.
2. (3 punti) Codificare la seguente parola: 01101101 (bit meno significativo a sinistra).
3. (2 punti) Spiegare come possa essere corretto un eventuale errore sul primo bit della parola data.

ESERCIZIO 5 (4 punti)

Implementare, tramite il set di istruzioni a singolo indirizzo fornito in tabella, l'espressione:

$$Z = A/B + C / (A+B)$$

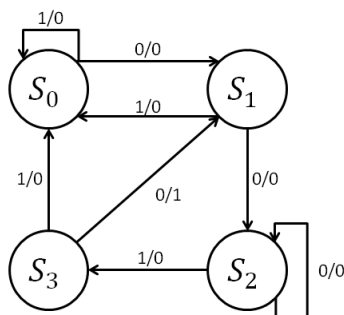
Istruzione	Semantica
LOAD X	$M[X] \rightarrow ACC$
STORE X	$ACC \rightarrow M[X]$
ADD X	$ACC \leftarrow ACC + M[X]$
DIV X	$ACC \leftarrow ACC / M[X]$

$M[X]$ indica il contenuto della locazione di memoria X.

Si usino esclusivamente le quattro locazioni A, B, C, Z indicate e si faccia attenzione a non alterare i valori conservati in A, B, C durante l'esecuzione delle istruzioni.

ESERCIZIO 1 (8 punti)

Risposta. Il grafo degli stati è dato da:



Che conduce alla seguente tabella delle transizioni dello stato, posto che ad ogni simbolo di stato si associ una coppia di bit in quest'ordine: S0=00, S1=01, S2=10, S3=11, che nel seguito indichiamo con la coppia AB. Indichiamo con Z l'uscita della rete, e con TA e TB gli ingressi di ciascuno dei flip flop rappresentanti i bit di stato A e B.

A	B	X	A'	TA	B'	TB	Z
0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	0
0	1	0	1	1	0	1	0
0	1	1	0	0	0	1	0
1	0	0	1	0	0	0	0
1	0	1	1	0	1	1	0
1	1	0	0	1	1	0	1
1	1	1	0	1	0	1	0

Semplificando le espressioni ottenute con le mappe di Karnaugh:

<div> <div>AB</div> <div>X</div> <div>00011110</div> <div>0</div> <div>1</div> <div> <div>1</div> <div>1</div> <div>1</div> </div> <div> $T_A = B\bar{X} + AB$ </div> </div>		<div> <div>AB</div> <div>X</div> <div>00011110</div> <div>0</div> <div>1</div> <div> <div>1</div> <div>1</div> <div>1</div> <div>1</div> </div> <div> $T_B = BX + AX + \bar{A}\bar{X}$ </div> </div>	
---	--	---	--

La rete di transizione dell'uscita è infine: $Z = AB\bar{X}$

ESERCIZIO 2 (8 punti)

Risposta 1.

Ogni blocco è formato da 16 parole $\rightarrow 2^4 \rightarrow 4$ bit per il campo offset.

La cache è costituita da 128 parole $\rightarrow 2^7$, organizzate in $2^3 \rightarrow 8$ linee di 16 parole. Campo Index pari a 3 bit.

La primaria è indirizzata con 2K parole $\rightarrow 2^{11} \rightarrow 11$ bit, da cui si evince che il campo TAG sarà pari a 4 bit. In sintesi:

< TAG 4 bit > < Cache Index 3 bit > < Offset 4 bit >

Risposta 2.

La domanda ci richiede di indicare l'indirizzo dei blocchi con C.I. pari a 4. Si riferisce cioè ad un block frame così configurato:

Block Frame							
TAG				Cache Index			
x3	x2	x1	x0	100			

Dove abbiamo indicato con x3, x2, x1, x0 i quattro bit del TAG. Abbiamo dunque sedici possibili blocchi, con TAG da 0 a 15, ottenibili moltiplicando il relativo valore del TAG per il numero delle 8 (2^3) linee della cache. Infatti in ogni linea della cache possiamo avere uno dei 16 sedici possibili blocchi, con TAG diverso. Da cui: 0, 8, 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120 ai quali si deve aggiungere la costante 4, relativo al valore dell'index. I valori richiesti sono dunque: 4, 12, 20, 28, 36, 44, 52, 60, 68, 76, 84, 92, 100, 108, 116, 124.

Risposta 3.

Si deve applicare la formula della gerarchia di memoria a due livelli: $T = T_c + (1 - H_c) * T_p$.

$$T = 4 + (1 - 0.95) * 40 = 4 + 2 = 6 \text{ ns}$$

ESERCIZIO 3 (7 punti)

Risposta 1.

1 MB = 2^{10} KB occupa 2^8 settori da 4 KB, ovvero 256 settori. Poiché ciascuna traccia contiene 64 settori, saranno necessarie 4 tracce contigue libere.

Risposta 2.

I parametri generali del disco sono:

Trot = 10 ms, Tlat = 5 ms.

Bisogna calcolare nell'ordine: tempo di posizionamento dall'ultima alla prima traccia $\rightarrow 127$ ms.

Latenza perché si possa scrivere il primo settore utile: 5 ms.

Scrittura dei primi 64 settori: Trot.

Per i successivi tre settori:

- Posizionamento alla traccia adiacente: 1 ms
- Scrittura della traccia: Trot.

Il tempo complessivo è pari a : $127 + 5 + 10 + 3 * (1 + 10) = 127 + 15 + 33 = 175 \text{ ms}$.

ESERCIZIO 4 (6 punti)

Risposta 1.

Si deve rispettare la formula $2^K \geq N+K+1$

Da cui $K = 4$.

Risposta 2.

Indichiamo i bit della parola come segue:

b_0	b_1	b_2	b_3	b_4	b_5	b_6	b_7
0	1	1	0	1	1	0	1

In accordo con lo schema che associa i bit di controllo ad un sottoinsieme di bit di parola controllati, si ha:

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

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

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

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

Le posizioni di ciascun bit di controllo precedono i relativi bit controllati, quindi la stringa è pari a:

000111011101

Dove in neretto abbiamo evidenziato la posizione dei bit di controllo dal meno significativo al più significativo.

Risposta 3.

Dalle formule precedenti, si evince che un errore su b_0 implica la commutazione dei bit da esso controllati, c_0 e c_1 . In ricezione il sistema si "accorge" di questo separando i bit di controllo ricevuti (rimasti inalterati) da quelli della parola; ricalcola i bit di controllo e verifica quali bit differiscano da quelli ricevuti. Come scritto, in questo caso verificherà la contemporanea commutazione di c_0 e c_1 . L'intersezione dei sottoinsiemi di bit controllati darà come risultato l'individuazione del bit errato, tramite l'operatore-distanza XOR.

ESERCIZIO 5 (4 punti)

Risposta. La sequenza è:

Istruzione	Semantica	Contenuto dell'accumulatore
LOAD A	$M[A] \rightarrow ACC$	$M[A]$
ADD B	$ACC \leftarrow ACC + M[B]$	$M[B] + M[A]$
STORE Z	$M[Z] \leftarrow ACC$	$M[B] + M[A]$
LOAD C	$M[C] \rightarrow ACC$	$M[C]$
DIV Z	$ACC \leftarrow ACC / M[Z]$	$M[C] / (M[B] + M[A])$
STORE Z	$M[Z] \leftarrow ACC$	$M[C] / (M[B] + M[A])$
LOAD A	$M[A] \rightarrow ACC$	$M[A]$
DIV B	$ACC \leftarrow ACC / M[B]$	$M[A] / M[B]$
ADD Z	$ACC \leftarrow ACC + M[Z]$	$M[A] / M[B] + M[C] / (M[B] + M[A])$
STORE Z	$M[Z] \leftarrow ACC$	$M[A] / M[B] + M[C] / (M[B] + M[A])$