

SOLUZIONI DELLA PROVA SCRITTA DEL CORSO DI CALCOLATORI ELETTRONICI

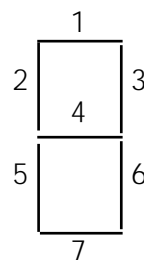
14 Settembre 2000

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

IMPORTANTE: Leggere la NOTA riportata a pag. 4 di questo documento

ESERCIZIO 1

Si progetti la rete logica che realizza un "visualizzatore a 7 segmenti" (ogni segmento è costituito da un led). Tale dispositivo consente di rappresentare le 10 cifre decimali, rappresentate in formato BCD, accendendo la combinazione opportuna di segmenti. Ipotizzare che ciascun segmento venga acceso attraverso il segnale 1 e venga mantenuto spento con il segnale 0.



Soluzione

La tabella di verità della rete logica richiesta si realizza semplicemente osservando che:

Per rappresentare la cifra...	...è necessario accendere i segmenti...	X_1	X_2	X_3	X_4	S_1	S_2	S_3	S_4	S_5	S_6	S_7
0 (0000)	1, 2, 3, 5, 6, 7	0	0	0	0	1	1	1	0	1	1	1
1 (0001)	3, 6	0	0	0	1	0	0	1	0	0	1	0
2 (0010)	1, 3, 4, 5, 7	0	0	1	0	1	0	1	1	1	0	1
3 (0011)	1, 3, 4, 6, 7	0	1	0	0	0	1	1	1	0	1	0
4 (0100)	2, 3, 4, 6	0	1	0	1	1	1	0	1	0	1	1
5 (0101)	1, 2, 4, 6, 7	0	1	1	0	0	1	0	1	1	1	1
6 (0110)	2, 4, 5, 6, 7	0	1	1	1	1	0	1	0	0	1	0
7 (0111)	1, 3, 6	1	0	0	0	1	1	1	1	1	1	1
8 (1000)	1, 2, 3, 4, 5, 6, 7, 8	1	0	0	1	1	1	1	1	0	1	0
9 (1001)	1, 2, 3, 4, 6	1	0	1	0	d	d	d	d	d	d	d
				⋮						⋮		
		1	1	1	1	d	d	d	d	d	d	d

La rete logica richiesta avrà dunque quattro ingressi (cioè la cifra BCD) e sette uscite, ciascuna delle quali comanda l'accensione di uno dei sette segmenti, come riportato nella tabella a destra.

Le mappe di Karnaugh che sintetizzano le funzioni minime per l'accensione dei sette segmenti sono le seguenti:

$S_1 = X_1 + X_2 X_4 + \bar{X}_2 X_3 + \bar{X}_2 \bar{X}_4$	$S_2 = X_1 + X_2 \bar{X}_3 + \bar{X}_3 \bar{X}_4 + X_2 \bar{X}_4$	$S_3 = \bar{X}_2 + X_3 X_4 + \bar{X}_3 \bar{X}_4$	$S_4 = X_1 + X_2 \bar{X}_3 + \bar{X}_2 X_3 + X_3 \bar{X}_4$
---	---	---	---

$S_5 = \bar{X}_2 \bar{X}_4 + X_3 \bar{X}_4$	$S_6 = \bar{X}_3 + X_4 + X_2$	$S_7 = X_3 \bar{X}_4 + \bar{X}_2 X_3 + \bar{X}_2 \bar{X}_4 + X_2 \bar{X}_3 X_4$
---	-------------------------------	---

ESERCIZIO 2

- Si consideri una memoria a disco con le seguenti caratteristiche: velocità di rotazione 5400 giri/min, tempo necessario a spostarsi da una traccia alla successiva pari a 3 ms, settori da 512 byte, 200 settori per traccia. Calcolare il tempo necessario per leggere un file di 100 kbyte ipotizzando che:
 - Il file sia stato registrato in settori consecutivi di una stessa traccia e la testina si trovi posizionata sul primo blocco del file.
 - La distanza fra due blocchi successivi del file si trovino a una distanza media pari a 3 tracce (N.B.: 1 "blocco" del file è registrato in 1 settore del disco).
- Si consideri una gerarchia di memoria a due livelli: cache e memoria primaria. Se il tempo di accesso alla memoria primaria è pari a T e quello relativo alla cache è pari a $T/5$, trovare il valore dell'*hit ratio* per cui il tempo di accesso medio alla gerarchia di memoria è pari a $2T/5$.

Soluzione

- $T_{rot} = 60/V_{rot} = 11,11 \text{ ms}$
 $T_{lat} = T_{rot}/2 = 5,56 \text{ ms}$
 $T_{trasf} \text{ (1 settore)} = T_{trasf} \text{ traccia} / (\text{n.ro settori}) = T_{rot}/200 = 55,6 \mu\text{s}$
 (a) Il file contiene $100\text{k}/512 = 200$ blocchi, dunque è interamente contenuto in una traccia. Il tempo di lettura coincide con il tempo di trasferimento (la testina è posizionata sul primo settore, dunque tempo di posizionamento e di latenza sono nulli) e questo coincide con il tempo di rotazione: $T_{lett \text{ file}} = T_{rot} = 11,11\text{ms}$
 (b) $T_{lett \text{ file}} = (100\text{k}/512) * (3 * 3 \text{ ms} + 5,56 \text{ ms} + 55,6 \mu\text{s}) = 2,932312 \text{ s}$
- L'equazione risolutiva è: $2T/5 = T/5 + (1-H)T$
 Da questa si ricava il valore di $H = 4/5 = 0.8$

ESERCIZIO 3

Si consideri una architettura basata su accumulatore. Il contenuto della memoria a un certo istante è il seguente (per semplicità usiamo la notazione decimale): l'indirizzo 100 contiene il valore 300; l'indirizzo 250 contiene 150; l'indirizzo 300 contiene 500; l'indirizzo 400 contiene 250. Dire quale sarà il contenuto dell'accumulatore dopo le seguenti istruzioni:

Istruzioni	Modalità indirizz. (1)	Modalità indirizz. (2)	Modalità indirizz. (3)
LOAD 100	immediato	diretto	diretto
ADD 400	diretto	diretto	indiretto

nei tre casi riportati nelle tre colonne.

Soluzione

Nel caso di indirizzamento immediato, il campo indirizzi contiene l'operando; nel caso di indirizzamento diretto il campo indirizzi contiene l'indirizzo di memoria ove risiede l'operando; infine nel caso di indirizzamento indiretto il campo indirizzi contiene l'indirizzo di memoria che a sua volta contiene l'indirizzo di memoria ove risiede l'operando.

Caso (1):

'LOAD 100' carica il valore 100 nell'accumulatore $AC \leftarrow 100$

'ADD 400' somma al contenuto dell'accumulatore il valore in memoria all'indirizzo 400:

$AC \leftarrow AC + M[400]$, $AC \leftarrow 100 + 250$. Il contenuto finale dell'accumulatore è 350

Caso (2):

'LOAD 100' carica nell'accumulatore il valore in memoria all'indirizzo 100: $AC \leftarrow M[100]$,
 $AC \leftarrow 300$

'ADD 400' somma al contenuto dell'accumulatore il valore in memoria all'indirizzo 400:

$AC \leftarrow AC + M[400]$, $AC \leftarrow 300 + 250$. Il contenuto finale dell'accumulatore è 550.

Caso (3):

'LOAD 100' carica nell'accumulatore il valore in memoria all'indirizzo 100: $AC \leftarrow M[100]$,
 $AC \leftarrow 300$

'ADD 400' somma al contenuto dell'accumulatore il valore in memoria all'indirizzo contenuto in 400:

$AC \leftarrow AC + M[M[400]]$, $AC \leftarrow 300 + M[250]$, $AC \leftarrow 300 + 150$. Il contenuto finale dell'accumulatore è 450.

ESERCIZIO 4

1. Date le seguenti partizioni di memoria: 100K, 500K, 200K, 300K e 600K (in ordine di indirizzo crescente) dire come vengono disposti dagli algoritmi first-fit e best-fit i processi di 212K, 417K, 112K e 426 K (nell'ordine). Dire quali di questi algoritmi utilizza la memoria nel modo più efficiente.
2. Risolvere l'esercizio nel caso in cui la memoria sia gestita con la tecnica della paginazione (non su richiesta) con pagine di 50K e confrontare il risultato con quello ottenuto nel caso di partizioni fisse.

Soluzione

1. La strategia first-fit alloca ciascun processo nella prima partizione libera che può contenerlo, esaminando le partizioni in ordine di indirizzo crescente. In questo caso le partizioni vengono scandite nell'ordine indicato nel testo. La strategia best-fit alloca ciascun processo nella prima partizione libera che può contenerlo, esaminando le partizioni in ordine di dimensione crescente. In questo caso le partizioni vengono scandite nell'ordine: 100K, 200K, 300K, 500K, 600K.

Algoritmo	100K	500K	200K	300K	600K
First-fit		212K	112K		417K
Best-fit		417K	112K	212K	426K

In questo caso utilizzando l'algoritmo di First-fit il processo di 426K non può essere caricato.

2. Se la memoria è gestita con la tecnica della paginazione con dimensione delle pagine pari a 50K, la memoria avrebbe a disposizione 34 pagine (1700K/50K). Il primo processo occupa 5 pagine, il secondo 9 pagine, il terzo 3 pagine e il quarto 9 pagine. In totale la memoria occupata è pari a 26 pagine e quella libera a 8 pagine (400 K). Questa soluzione consente di avere una minore frammentazione della memoria rispetto al caso best-fit e, a maggior ragione, first-fit.

ESERCIZIO 5

Mostrare come si possa realizzare un pipeline aritmetico per effettuare la somma di un vettore formato da numeri in virgola mobile. Calcolare il tempo necessario ad eseguire la somma $A_i + B_i$ ($i = 1, \dots, 50$) utilizzando un'architettura con pipeline aritmetico, ipotizzando che ciascuno stadio richieda 15 ns. Spiegare il ragionamento seguito per calcolare il risultato

Soluzione

Un pipeline aritmetico può essere realizzato con quattro stadi: confronto degli esponenti, allineamento delle mantisse, somma, normalizzazione del risultato. La somma di 50 elementi avviene nel seguente modo: per calcolare la prima somma occorre un tempo pari a quattro stadi, perché ipotizziamo che inizialmente la pipeline sia vuota. I risultati delle somme successive vengono prodotti a distanza di 15ns (il tempo necessario a completare uno stadio) perché il pipeline funziona a regime. Dunque il tempo complessivo necessario è pari a: $4 \cdot 15 + 49 \cdot 15 = 795 \text{ ns}$.

NOTA	Esercizio 1	Esercizio 2	Esercizio 3	Esercizio 4	Esercizio 5
Vecchio Ordinamento	8 punti	1.a : 2 punti 1.b: 3 punti 2: 2 punti	5 punti	dom. 1: 5 punti dom. 2: 3 punti	5 punti
Nuovo Ordinamento	10 punti	1.a : 3 punti 1.b: 4 punti 2: 2 punti	7 punti	Solo dom. 1: 7 punti	Assente