

SOLUZIONI DELLA PROVA SCRITTA DEL CORSO DI
CALCOLATORI ELETTRONICI
NUOVO ORDINAMENTO DIDATTICO
29 Settembre 2000

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

ESERCIZIO 1 (8 punti)

Si realizzi una ALU che esegua le seguenti 8 operazioni su due operandi A e B a n bit:

$$A + B, A, \bar{B}, A + \bar{B}, A + B + 1, A + 1, \bar{B} + 1, A - B$$

Soluzione

Per realizzare le 8 operazioni richieste si può utilizzare un parallel adder e una opportuna rete logica. Si può osservare che le ultime quattro operazioni si ottengono dalle prime 4 sommando 1, cioè imponendo che il c_{in} del parallel adder sia uguale a 1. Pertanto con una rete logica con due variabili di selezione, s_1 ed s_0 , si realizzano gli ingressi al parallel adder per le prime quattro operazioni. Le 8 operazioni possono essere ottenute secondo la tabella seguente:

s_1	s_0	$C_{in} = 0$	$C_{in} = 1$
0	0	$F = A + B$	$F = A + B + 1$
0	1	$F = A$	$F = A + 1$
1	0	$F = \bar{B}$	$F = \bar{B} + 1$
1	1	$F = A + \bar{B}$	$F = A - B = (A + \bar{B} + 1)$

Siano X e Y le due uscite della rete logica che ha in ingresso A , B , s_1 ed s_0 . X e Y sono anche gli ingressi del parallel adder. Affinché l'uscita del parallel adder fornisca i risultati delle operazioni richieste, dovrà essere:

s_1	s_0	X_i	Y_i
0	0	A_i	B_i
0	1	A_i	0
1	0	0	\bar{B}_i
1	1	A_i	\bar{B}_i

Le equazioni di X_i e Y_i in funzione di s_1 , s_0 , A_i e B_i sono le seguenti:

$$X_i = \bar{s}_1 A_i + s_0 A_i$$

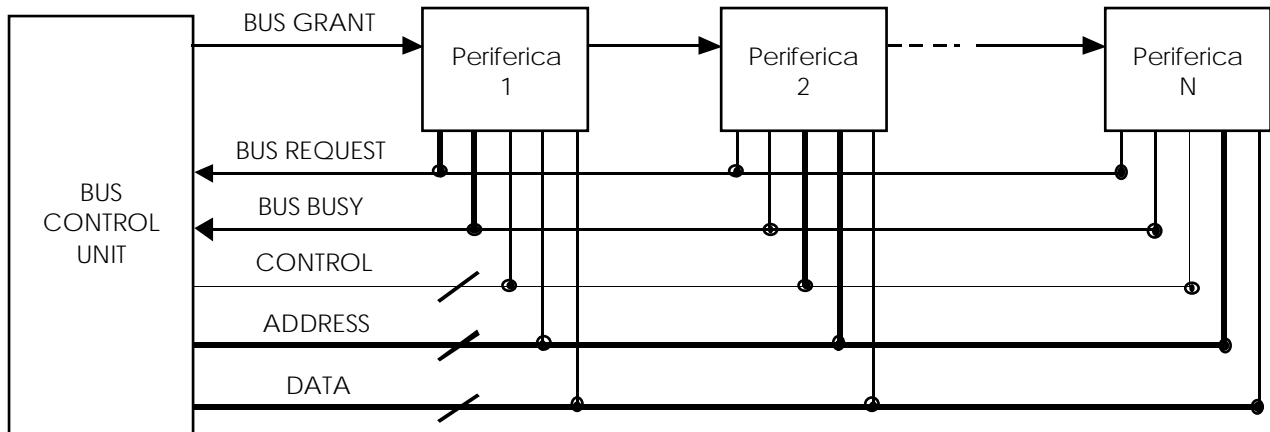
$$Y_i = s_1 \bar{B}_i + \bar{s}_1 \bar{s}_0 B_i$$

ESERCIZIO 2 (8 punti)

Un certo numero di periferiche sono collegate al calcolatore per mezzo di un bus sincrono. L'arbitraggio viene effettuato secondo lo schema "daisy chain".

1. Mostrare lo schema di collegamento delle periferiche al bus indicando le diverse linee che devono essere presenti nel bus per gestire l'arbitraggio e il trasferimento dati (4 punti)
2. Descrivere il protocollo usato dal controller del bus per individuare la periferica che ha richiesto l'uso del bus (2 punti)
3. Descrivere il protocollo per effettuare la lettura di un dato dalla memoria iniziato dalla periferica su un bus sincrono. (2 punti)

Soluzione



1. Le linee necessarie per il bus sono indicate in figura. Le linee di bus request, grant e busy, sono utilizzate per effettuare l'arbitraggio, mentre le linee control, address e data vengono usate per trasferire i dati. N.B. le linee di control, address e data sono in realtà "gruppi" di linee. La motivazione è contenuta nella risposta 3.
2. Arbitraggio in daisy chain: La periferica invia il segnale di richiesta del bus sulla linea bus request, se la linea bus busy indica che il bus è libero. Il controllore del bus invia in risposta il segnale di bus grant sull'apposita linea. Questo segnale viene propagato serialmente fra tutte le periferiche, fino a raggiungere la periferica che ha fatto richiesta, che blocca la propagazione del segnale di bus grant e attiva il segnale di bus busy. (in caso di richieste contemporanee la propagazione del bus grant viene bloccata dalla prima periferica che ha fatto richiesta). A questo punto la periferica può trasferire i dati.
3. Sulle linee di controllo viene inviato il segnale sul tipo di trasferimento, in questo caso lettura (ci sono più linee, ciascuna per un tipo di trasferimento come lettura, scrittura, trasferimento di blocchi, ecc.), e contemporaneamente l'indirizzo di memoria che deve essere letto sulla linea indirizzi (address). Trascorso il tempo di ciclo della memoria, il dato viene inviato sulla linea data che può consentire il trasferimento contemporaneo di un certo numero di byte.

ESERCIZIO 3 (9 punti)

Si ipotizzi che un processo sia suddiviso in quattro segmenti di dimensione uguale, e che il sistema operativo, per ciascun segmento, predisponga una "page map table" con 8 locazioni. In altre parole il sistema operativo implementa una memoria virtuale segmentata e paginata. Si sappia inoltre che la dimensione di ciascuna pagina sia uguale a 2 kbyte.

1. Qual è la dimensione massima di ciascun segmento? (2 punti)
2. Qual è la dimensione dello spazio degli indirizzi logici? (3 punti)
3. Se il sistema a un certo istante accede alla locazione di memoria di indirizzo fisico 00021ABC, qual è il formato dell'indirizzo logico che ha generato questo indirizzo fisico? (4 punti)

Soluzione

1. Ciascun segmento può contenere al massimo 8 pagine da 2 kbyte ciascuna, dunque la sua dimensione massima è pari a 16 kbyte.
2. L'indirizzo logico nel caso di memoria segmentata e paginata, con 4 segmenti, 8 pagine per segmento e pagine di dimensione 2 kbyte è formato da 16 bit, secondo lo schema seguente:

#segmento	#pagina all'interno del segmento	#byte all'interno della pagina
2 bit	3 bit	11 bit

3. La corrispondenza fra indirizzo logico e indirizzo fisico viene effettuata dal sistema operativo come segue: i due bit più significativi servono a individuare il segmento nella segment map table. In questa tabella, nel caso di memoria segmentata e paginata, si trova l'indirizzo della PMT relativa al segmento indirizzato dai 2 bit più significativi dell'indirizzo logico. Individuata la PMT, in corrispondenza dell'indirizzo logico di 3 bit si trova l'indirizzo fisico della pagina indirizzata. La parola cercata si trova nella pagina fisica usando gli 11 bit meno significativi dell'indirizzo logico. Nell'esempio dell'esercizio, gli 11 bit meno significativi (cioè 01010111100 indirizzano il byte all'interno della pagina), mentre i restanti bit indicano il numero di pagina fisica nella memoria, che si ricava come spiegato sopra.

ESERCIZIO 4 (8 punti)

Si abbiano a disposizione 32 bit per rappresentare numeri frazionari in virgola mobile nel formato IEEE 754 con esponente di 8 bit in eccesso 127 e mantissa frazionaria e normalizzata rappresentata in segno e valore.

- (a) Calcolare il massimo e il minimo numero positivo rappresentabile; (2 punti)
 (b) Mostrare quali sono i passi fondamentali dell'algoritmo utilizzato dal calcolatore per effettuare la somma fra due numeri frazionari rappresentati in virgola mobile. Eseguire la somma fra i seguenti numeri (senza dimenticare il bit implicito!) usando l'algoritmo descritto (4 punti):
 0 110100000000000000000000 10000000
 0 001000000000000000000000 01111110
 (c) Verificare il risultato ottenuto al punto precedente, trasformando i numeri in decimale. (2 punti)

Soluzione

1. Il "range" dell'esponente è: $-127 \div +128$. La mantissa è di $32 - 8 - 1 = 23$ bit
 Il massimo numero positivo rappresentabile è $(1 + 1 - 2^{-23}) \cdot 2^{+128}$
 Il minimo numero positivo rappresentabile è $(1) \cdot 2^{-127}$
 N.B. La mantissa normalizzata nel formato IEEE 754 è del tipo 1.M: il valore massimo si ottiene con tutti i 23 bit di M pari a 1 mentre il minimo con tutti i bit di M pari a 0
2. Per sommare due numeri in virgola mobile i passi necessari sono i seguenti:
- trasformazione del numero più piccolo in modo che l'esponente sia uguale a quello del numero più grande (la mantissa del numero più piccolo risulta dunque denormalizzata)
 - somma delle mantisse
 - eventuale normalizzazione del risultato
- Nell'esempio riportato l'esponente del primo numero è +1 e l'esponente del secondo numero è -1. Dunque devo effettuare lo scalamento della mantissa del primo numero di 2 posizioni a destra in modo che il campo esponente del secondo numero valga +1:
- 0 001000000000000000000000 01111110 → 0 010010000000000000000000 10000000
- Adesso si può eseguire la somma delle mantisse
- $1.110100000000000000000000 + 0.010010000000000000000000 = 10.0001100000000000000000$
 (N.B. il bit implicito ha valore diverso a causa della denormalizzazione del secondo numero)
- La mantissa risultante non è normalizzata. Per normalizzare occorre effettuare uno scalamento della mantissa verso destra e incrementare l'esponente di una unità. Il risultato finale è 0 000011000000000000000000 10000001
3. Il primo numero assegnato ha il seguente valore decimale:
 $(1 + 2^{-1} + 2^{-2} + 2^{-4}) \cdot 2 = 3.625$
 Il secondo numero: $(1 + 2^{-3}) \cdot 2^{-1} = 0.5625$
 La somma fra questi due numeri decimali è 4.1875, che coincide con il risultato calcolato in precedenza con l'algoritmo di somma $(1 + 2^{-5} + 2^{-6}) \cdot 2^2 = 4.1875$