

Architettura degli Elaboratori, 2008-09

Appello del 1 giugno 2009

Domanda 1

Una unità di elaborazione U

- contiene un componente logico memoria M di capacità 64 parole ognuna di 32 bit;
- riceve in ingresso, da una stessa unità U_M , messaggi (A, B, C, J) con A e B parole di 32 bit, C di 64 bit, e J di 4 bit. I valori di C hanno sempre la seguente caratteristica: uno ed un solo bit vale 1 e tutti gli altri 0; nel seguito con $P(C)$ si indicherà la posizione del bit uguale a 1 in C ;
- invia in uscita messaggi di una parola ad una delle 16 unità U_0, \dots, U_{15} .

Il funzionamento è il seguente:

- se $A < B$: scrive nella locazione di M di indirizzo uguale a $P(C)$ il contenuto della stessa locazione incrementato di 1, ed invia il valore scritto all'unità U_J ;
- se $A = B$: scrive nella locazione di M di indirizzo uguale a $P(C)$ il contenuto della stessa locazione incrementato di 1;
- se $A > B$: invia il contenuto della locazione di M di indirizzo uguale a $P(C)$ incrementato di 1 all'unità U_J .

È noto il ritardo t_p di una porta logica con al più 8 ingressi. Il ritardo di una ALU vale $5t_p$.

Spiegare e mostrare in dettaglio la struttura del componente logico memoria M in modo da utilizzare direttamente il valore di C per l'indirizzamento.

Scrivere e spiegare il microprogramma e valutare il tempo di elaborazione di U , in funzione di t_p , rispettando entrambi i seguenti requisiti:

- a) minimizzare il numero di variabili di condizionamento,
- b) minimizzare il numero di cicli di clock.

Mostrare la struttura della Parte Operativa.

Domanda 2

Si dica se le seguenti affermazioni sono vere o false, commentando adeguatamente la risposta:

- a) data una rete sequenziale nella quale il registro di stato è di un solo bit, esiste sempre una rete combinatoria che presenta un comportamento (corrispondenza fra ingressi ed uscite) equivalente;
- b) processi con una memoria virtuale di dimensione MV non possono essere eseguiti efficientemente su un'architettura con cache di dimensioni $MC < MV$;
- c) la compilazione di un programma in codice assembler richiede di determinare esattamente la mappa d'uso della memoria virtuale.

Domanda 3

Una unità di I/O collegata ad una CPU D-RISC può segnalare con una interruzione un certo evento, al quale corrisponde una informazione consistente in una sequenza di M interi, dove il valore M è comunicato di volta in volta dal dispositivo associato all'unità.

Lo handler dell'evento deve sommare gli M interi e memorizzare il risultato in una locazione condivisa tra tutti i processi secondo il modello degli indirizzi logici coincidenti.

Spiegare come progettare lo handler, e scrivere lo handler in D-RISC.

Cenno di soluzione

Domanda 1

Il microprogramma consta di una sola microistruzione, e il tempo medio di elaborazione è uguale a un ciclo di clock. Le variabili di condizionamento si riducono a due (RDYM da U_M e ACKIN[J] da $U[J]$), utilizzando intensivamente controllo residuo:

- semplici combinazioni delle variabili booleane *segno(A-B)* e *zero(A-B)* permettono di controllare la scrittura nella memoria M e nell'interfaccia verso $U[J]$,
- RDY, ACK e registri di uscita verso le U_J sono controllate, mediante commutatori/selezionatori, dal valore J.

In realtà, anche le variabili di condizionamento RDYM e e ACKIN[J] possono essere eliminate, condizionando ulteriormente la scrittura in qualunque registro mediante l'and di tali indicatori.

Il commutatore di uscita e il selezionatore d'ingresso della memoria M sono realizzati utilizzando le variabili di controllo in forma completamente decodificata all'uscita di C: per il commutatore di uscita, la i -esima delle 64 porta AND ha in ingresso l'uscita dell' i -esimo registro-cella ed il valore di $C[i]$.

Domanda 2

- falso. Il fatto che il registro di stato contenga un solo bit indica che ci sono due stati interni, dunque esiste uno stato e non possiamo utilizzare una rete combinatoria.
- falso. La dimensione della cache deve essere in grado di contenere l'insieme di lavoro (working set) del processo per avere un'esecuzione efficiente, non l'intero spazio di memoria virtuale.
- vero. Serve al compilatore per determinare gli indirizzi utilizzati per tutte le istruzioni e dati del programma.

Domanda 3

Poiché la sequenza non può essere inviata mediante il messaggio di interruzione, lo handler deve ricavare l'indirizzo della sequenza; questa può risiedere nello spazio fisico di I/O se si può usare solo il MMI/O. L'indirizzo della sequenza può essere noto a priori a tutti i processi, oppure inviato dall'unità di I/O, in quanto viene fatto uso del metodo con indirizzi logici coincidenti. Lo stesso metodo viene utilizzato per determinare l'indirizzo in cui scrivere il risultato da parte di qualunque processo esegua lo handler.

La struttura condivisa che contiene la sequenza riporta, nella prima posizione, il valore M .