

Architettura degli elaboratori – A.A. 2015-2016

Secondo appello sessione estiva – 5/7/2016

Si riporti su ciascun foglio consegnato Nome, Cognome Numero di matricola e corso (A o B).
I risultati e i calendari degli orali saranno pubblicati sulle pagine web didawiki e/o dei docenti appena disponibili.
Gli orali si svolgeranno immediatamente dopo la pubblicazione dei risultati.

Domanda 1.

Dato un vettore X di N posizioni, l'operazione di convoluzione

- sostituisce ad ogni posizione i diversa dalla prima e dall'ultima la media aritmetica dei valori x_i, x_{i-1} e x_{i+1}
- calcola la somma di tutte le differenze fra nuovi e vecchi valori nelle diverse posizioni del vettore.

La convoluzione è ripetuta fino a quando la somma delle differenze scende sotto una certa soglia. Lo pseudo codice può essere il seguente:

```
do {
    diff = 0;
    for(int i=1; i<N-2; i++) {
        new = (x[i-1]+x[i]+x[i+1])/3;
        diff = diff + (new-x[i]);
        x[i]=new;
    }
} while(diff > threshold);
```

Si scriva il codice D-RISC che calcola l'operazione di convoluzione su vettori di interi e si valuti il tempo di completamento della sola parte in corsivo nello pseudocodice (una iterazione del `do-while`) su due tipi di processore:

- un processore D-RISC pipeline, scalare, con EU slave che calcola una divisione fra interi in $4t$
- un processore D-RISC pipeline con multithreading simmetrico a due vie e con la stessa unità EU slave del caso precedente. In questo caso si assuma che i due thread eseguendo i calcoli della metà inferiore e superiore del vettore, rispettivamente

discutendone le eventuali differenze o similitudini.

Domanda 2.

Si consideri un pipeline di quattro unità firmware U_1, U_2, U_3, U_4 , implementate su un unico chip. Ciascuna delle U_i riceve un singolo dato da 32 bit da U_{i-1} , elabora un risultato ancora da 32 bit in k_i cicli di clock ($\sum k_i = K$ e $\forall i: k_i \geq 2\tau$) e manda il risultato a U_{i+1} . Il risultato è calcolato da ciascuna delle U_i mediante k_i microoperazioni $\mu op_1, \dots, \mu op_{k_i}$ e vale che per qualunque $i \in [1, k_i]$

$$W(\mu op_{i-1}) \cap R(\mu op_i) \neq \emptyset$$

U_1 riceve valori in ingresso da un'unità esterna, con un tempo di interarrivo $T_a = 2\tau$, e U_4 manda risultati ad una unità esterna sempre pronta a riceverli. Si calcoli (fornendo tutte le necessarie motivazioni) il tempo di completamento necessario al pipeline per produrre i risultati relativi a m ingressi.

Domanda 3.

Dato il microcodice:

- (RDY=0) nop,
(=1) $(IN+I)\%N \rightarrow I, I \rightarrow J$, reset RDY, set ACK, 1
- (segno($M[(I-1)\%N] - M[(I+1)\%N]$) = 0) $M[I] \rightarrow M[J]$, 0
(=1) $M[J] \rightarrow M[I]$, 0

si discutano le caratteristiche della memoria M dettate dalla struttura del micro codice.