

Architettura degli elaboratori—A. A. 2016-2017—2° Appello—7 febbraio 2017

Riportare nome, cognome, numero di matricola e corso (A/B) in alto a destra su tutti i fogli consegnati.
I risultati e il calendario degli orali saranno pubblicati su WEB appena disponibili.

Domanda 1

Si consideri un sistema costituito da 8 unità firmware (U_1, \dots, U_8) collegate in pipeline: l'unità U_i riceve dati da U_{i-1} e invia risultati alla unità U_{i+1} . L'unità U_1 riceve dati da una unità U_a mentre la U_8 invia i propri risultati ad una unità U_b . Ciascuna U_i ha al suo interno una memoria M con 2^{13} posizioni da 32 bit. Ogni interfaccia fra due unità permette l'invio di una terna di valori interi X , I e OP , rispettivamente da 32, 16 e 2 bit. I 16 bit di I vengono interpretati come 3 bit di indirizzo unità (parte più significativa) e 13 bit di indirizzo da utilizzare per accedere alla memoria interna M . Ciascuna delle unità U_j esegue una delle seguenti operazioni esterne:

- se $OP = 00$ e i bit di indirizzo unità di I valgono j , il valore X in ingresso viene memorizzato alla posizione di M determinata dai 13 di indirizzo di I e il vecchio valore di tale posizione viene trasmesso alla unità U_{j+1} , utilizzando come valore I lo stesso ricevuto in ingresso e come valore OP il valore 10; se i bit di indirizzo unità di I non valgono j , I , X e OP vengono passati all'unità successiva
- se $OP = 01$ e i bit di indirizzo unità di I valgono j , all'unità U_{j+1} viene inviato il valore in memoria M alla posizione determinata dai 13 bit di indirizzo di I , utilizzando lo stesso valore I quello ricevuto in ingresso e come valore OP il valore 10; se i bit di indirizzo unità di I non valgono j , I , X e OP vengono passati all'unità successiva
- se $OP = 10$ l'unità passa all'unità successiva senza modificarli i valori appena ricevuti per X , I e OP

L'unità U_a invia ad U_1 solo richieste per operazioni $OP=00$ o 01 . Si vuole conoscere la latenza, in t_p , richiesta dall'esecuzione di una operazione ordinata da U_a considerando come esecutore dell'operazione l'intero pipeline formato dalle 8 unità.

Domanda 2

Si consideri un sistema costituito dal processore D-RISC sequenziale (quello dell'interprete firmware) e dotato di una gerarchia di memoria che comprende un'unico livello di cache, comune per dati e istruzioni, set associativa su insiemi (4 linee per insieme), con $\sigma = 16$ e 1K insiemi. La memoria principale è modulare, interallacciata, con 4 moduli da 1G parole ciascuno e $\tau_M = 400\tau$ (con τ ciclo di clock della CPU). Il t_{tr} fra memoria principale e cache è pari a 4τ .

Si consideri quindi il codice D-RISC che deriva dalla compilazione dello pseudo codice

```
int N = 1024, a[N], b[N], c[N];
for(int i=0; i<N; i++) {
    if(a[i] == b[i]) c[i] = a[i]*4;
    else             c[i] = a[b[i]%N]*2;
}
```

e si calcolino:

- il numero di fault generati dal programma
- la traccia degli indirizzi generati dal programma e quelli utilizzati per l'accesso alla cache durante la prima iterazione del for, indicando per ciascun indirizzo il numero di insieme nella cache
- il tempo di completamento della prima iterazione, assumendo che $A[0]$ sia diverso da $B[0]$

assumendo che:

- i vettori A , B e C e il codice siano allocati rispettivamente dal compilatore agli indirizzi 1K, 2K, 3K e 0
- che le pagine in memoria centrale siano di 2K parole
- che $\text{tabril}[] = \{<12,1>, <6,1>, <24,1>, <1,1>, \dots\}$ dove ogni coppia è data da $<IPF, \text{bit di presenza}>$