

# CALCOLATORI ELETTRONICI M

## ARCHITETTURA DEI CALCOLATORI ELETTRONICI M

08/01/2024 – Esercizio 1 & 2

**Tempo disponibile: 150 minuti**

### Esercizio 1 - Descrizione Sistema

Un RV32IMAFD con  $T_{CK} = T$  dispone di tre unità funzionali, A, M e D “multiciclo” capaci di eseguire le seguenti istruzioni su operandi in virgola mobile:

**A: FADD (in 2T)      M: FMUL (in 3T)      D: FDIV (in 4T).**

Si consideri il seguente frammento di codice e si faccia l’ipotesi che in T1 si abbia  $F_i = i$  per ogni valore di  $i$  compreso tra 0 e 31:

```

fdiv.s f1, f10, f2
fdiv.s f6, f1, f5
fadd.s f6, f2, f3
fadd.s f3, f8, f2
fdiv.s f4, f6, f1
fmul.s f6, f10, f5

```

1. Si disegni il grafo delle dipendenze e si deduca il numero minimo di periodi di clock necessario a eseguire il codice assegnato tenendo conto solo delle dipendenze trovate (**1 punto**).
2. Si mostri la dinamica dell’esecuzione nel caso della CPU considerata nel punto 1 con 1 CDB, uno stadio di IF e uno di ID, e 2 RS per ognuna delle tre unità funzionali (*si ipotizzi che, in caso di conflitti sul CDB, la fase di WB dell’unità D abbia la priorità sulle altre*) (**3 punti**).
3. E’ possibile ridurre il numero di periodi di clock necessario ad eseguire il codice assegnato rispetto al valore risultante dalla risposta al punto 3? Si punti al numero minore possibile di periodi di clock e al numero minimo di modifiche, supponendo di poter apportare solo le seguenti modifiche all’architettura:
  - a. aumento di RS da due a tre in una o più unità funzionali
  - b. raddoppio o triplicazione del CDB per poter eseguire due o tre fasi di WB per clock
  - c. aggiunta di una o più unità funzionali con due RS

*In caso di incremento di unità funzionali, si ipotizzi di fare lo scheduling a rotazione sulle due unità funzionali uguali.*

Quali modifiche converrebbe apportare? Si motivi la risposta e si disegni la nuova dinamica di esecuzione. (**3 punti**).

4. Si disegni il film del registro f6 nel caso della dinamica di esecuzione di cui al punto 4. (**3 punti**)

### Esercizio 2 - Descrizione Sistema

Un sistema multicore S è composto da 2 cores RV32IMAC dotati ciascuno di una cache L1 dati privata a 2 vie da 8 KiB e linee da 32 bytes, gestita con stato MESI e con politica di scrittura Write Allocate in caso di miss di scrittura.

Nel sistema S esegue un’applicazione parallela composta da due tasks, T0 e T1, operanti su due array A, B, contenenti ognuno 512 elementi in virgola mobile a doppia precisione e su due variabili C, D in virgola mobile a doppia precisione.

- Il task T0 calcola il prodotto scalare tra A e B e salva il risultato in C ( $C = \sum_i \{B[i] * A[i]\}$ ).
- Il task T1 calcola la media del vettore A e salva il risultato in D ( $D = \sum_i \{A[i]\} / 512$ ).

```

void T0(double *A, double *B, double *C,
int n) {
    *C = 0;
    for (int i=0; i<n; i++) {
        *C += A[i] * B[i];
    }
}

void T1(double *A, double *B, double *D,
int n) {
    *D = 0;
    for (int i=0; i<n; i++) {
        *D += A[i];
    }
    *D = *D/n;
}

```

Si assume che i vettori A e B siano memorizzati a partire dall'indirizzo 0x0000\_1000 e 0x0000\_2000 rispettivamente e che le variabili C e D siano memorizzate a partire dall'indirizzo 0x0000\_3000 e 0x0000\_4000 rispettivamente. Si considerino altresì i seguenti codici ASM RISC-V che implementano il body dei due loop di calcolo dei tasks T0 e T1.

```

; Per entrambi i task T0 e T1 valgono le seguenti inizializzazioni:
; - x10 (e x15) contiene l'indirizzo dell'i-esimo elemento del vettore A
; - x11 contiene l'indirizzo dell'i-esimo elemento del vettore B
; - x12 contiene l'indirizzo della variabile
; --- C per il task T0
; --- D per il task T1
; - x13 contiene la dimensione dei vettori A e B

```

T0:	T1:
sd zero,0(x12)	sd zero,0(x12)
beq x13,zero,.L1	beq x13,zero,.L6
add x15,x10,x0	add x15,x10,x0
slli x13,x13,3	slli x13,x13,3
add x10,x10,x13	add x10,x10,x13
.L3:	.L7:
fld f14,0(x15)	fld f15,0(x12)
fld f15,0(x11)	fld f14,0(x15)
fmul.d f14,f14,f15	fadd.d f15,f15,f14
fld f15,0(x12)	fsd f15,0(x12)
fadd.d f15,f15,f14	addi x15,x15,8
fsd f15,0(x12)	bne x15,x10,.L7
addi x15,x15,8	.L6:
addi x11,x11,8	fcvt.d.lu f14,x13
bne x15,x10,.L3	fld f15,0(x12)
.L1:	fddiv.d f15,f15,f14
ret	fsd f15,0(x12)
	re

Quesiti:

1. Facendo riferimento al sistema S, si completi la mappa della memoria con i vettori A, B e variabili C e D
  - 1.1. Indicare l'indirizzo del LSB e MSB del primo e ultimo elemento dei vettori A, B, oltre all'LSB e MSB del primo ed ultimo byte di C e D. (**2 punti**)
  - 1.2. Specificare il numero e il valore dei bit di TAG e Set ID del primo ed ultimo elemento dei vettori A, B e delle variabili C e D (**2 punti**)
  - 1.3. Quanta memoria occupano complessivamente i vettori A, B e le variabili C e D? È possibile che tutti e due i vettori e le variabili si trovino contemporaneamente ed interamente all'interno delle cache del solo core 0 (**1 punto**)
2. Si analizzi la dinamica della cache dati assumendo che tutte le linee di cache dei due core siano inizialmente invalide e i bit LRU inizializzati a 0. Si assume che ciascuna iterazione del core1 esegua dopo l'iterazione i-1 del core0.
  - 2.1. Si mostri la dinamica del contenuto della cache, dello stato MESI e del valore del bit LRU durante l'esecuzione del primo loop della cache. (**3 punti**).
  - 2.2. Si mostri il contenuto della cache (per la prima e l'ultima linea di cache), lo stato MESI e il valore del bit LRU al termine del programma. (**3 punti**).
  - 2.3. Calcolare il numero di accessi, HIT, MISS, HIT rate, MISS rate e cicli di WB per ciascun core al termine dell'esecuzione del programma (**3 punti**).
  - 2.4. Si descriva in modo qualitativo come si modificherebbe l'analisi al punto 3 se la variabile D fosse stata allocata all'indirizzo 0x0000\_3010? (**2 punti**)