

Cognome e nome dello studente:

Matricola:

A.A. 2008-2009 – Appello del 23 Luglio 2009

1) [5] La pipeline disegnata sotto, contenente i circuiti per la propagazione, funziona correttamente anche per le istruzioni di sw, slt e addi? Motivare la risposta ed eventualmente modificarla in modo che le tre istruzioni di cui sopra vengano gestite correttamente.

2) [7] Si consideri il seguente frammento di codice:

```
lw $1,40($6)
beq $2,$0,Label ; Si supponga $2 == $0
sw $6,50($2)
Label: add $2,$3,$4
      sw $3,50($4)
```

Si supponga che non ci siano hazard sul controllo (predizione corretta delle branch) e che sia disponibile una sola memoria, non ci sia cioè distinzione tra memoria istruzioni e memoria dati. In questo caso si verifica un hazard strutturale ogni volta che viene prelevata un'istruzione nello stesso ciclo in cui si accede ad un dato in memoria. In questo caso, occorre privilegiare il prelievo dell'istruzione o del dato? Perché? Quali tipi di hazard (sui dati, sul controllo o strutturali) si possono eliminare aggiungendo delle nop? Cosa sono? Come funzionano? Eliminano i problemi in questo caso? Perché? Quale sarà il tempo complessivo di un'esecuzione corretta del codice riportato sopra?

3) [4] Cos'è un hazard? Uno stallo? Cos'è una pipeline multiple-issue (ad esecuzione parallela)? Che differenza c'è tra una pipeline multiple-issue statica e dinamica? Come funziona la fase di fetch nelle pipeline ad esecuzione parallela? Di che tipo è la pipeline della GPU? Cosa si intende per pipeline superscalare?

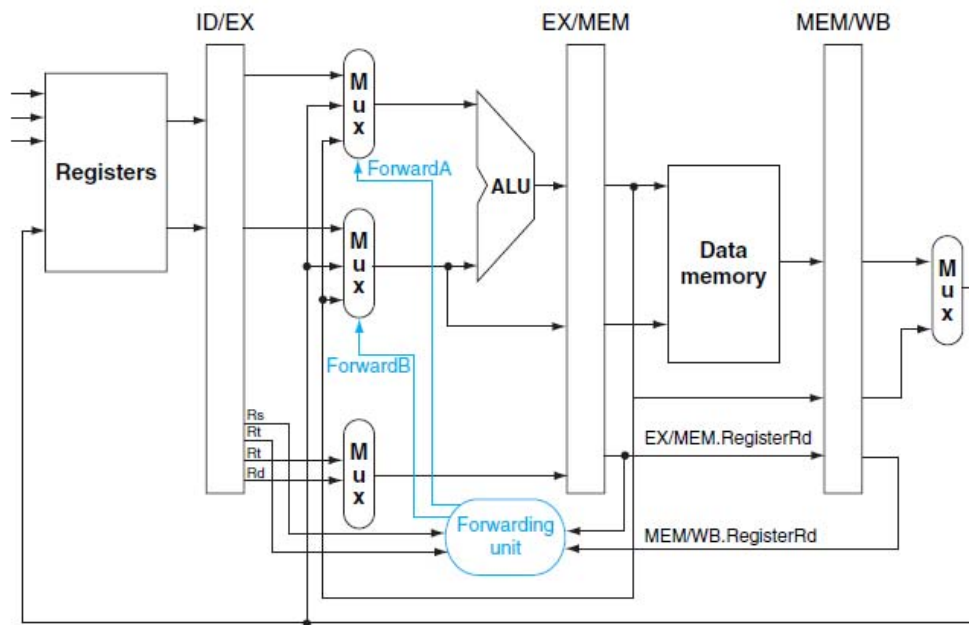
4) [6] Come vengono gestite eccezioni ed interrupt? Descrivere a grandi linee come vengono descritti da un'architettura MIPS e da un'architettura Intel. Supponendo che la risposta agli interrupt sia vettorizzata (cosa vuol dire?), su 4 word, come verrà memorizzato il seguente programma di risposta ad interrupt? E' necessario modificare il codice? Come?

```
add $s0$, $s1, $s2
mfco $s1, $13
mfco $s2 $14
addi $t0, $0,0
sub $s0, $s1, $s2
.....
```

5) [6] A cosa serve la memoria? Disegnare una cache a piacere associativa a 2 vie, il suo circuito di lettura e scrittura completo. Quali sono i pro e contro alle seguenti modifiche: a) aumentare la lunghezza della linea. b) diminuire la lunghezza della linea. c) Aumentare il grado di associatività. d) diminuire il grado di associatività

6) [5] Cos'è una gerarchia di memoria? Una gerarchia di bus? Perché si parla di gerarchia? C'è qualcuno che comanda? Come si interfaccia una periferica ad un bus (hardware / software)?

7) [2] Descrivere struttura e funzionamento di un disco rigido. Come si calcola e come si misura il tempo di accesso? Ed il tempo di trasferimento? Qual è il fattore limitante alle sue performance? Questo fattore porta a limitazioni sulla capacità e/o sulla velocità? Spiegare.



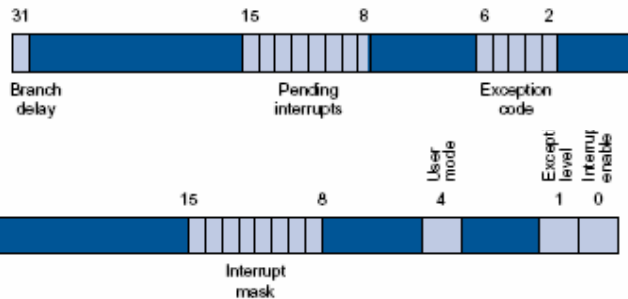
Register File

0	zero constant 0	16	s0 callee saves
1	at reserved for assembler	... (caller can clobber)	
2	v0 expression evaluation &	23	s7
3	v1 function results	24	t8 temporary (cont'd)
4	a0 arguments	25	t9
5	a1	26	k0 reserved for OS kernel
6	a2	27	k1
7	a3	28	gp Pointer to global area
8	t0 temporary: caller saves	29	sp Stack pointer
...	(callee can clobber)	30	fp frame pointer (s8)
15	t7	31	ra Return Address (HW)

Coprocessore 0

Nome del registro	Numero del registro in coprocessore 0	Utilizzo
Bad/Addr	8	Registro contenente l'indirizzo di memoria a cui si è fatto riferimento
Count	9	Timer
Compare	11	Valore da comparare con un timer. Genera un interrupt.
Status	12	Maschera delle interruzioni e bit di abilitazione. Stato dei diversi livelli di priorità (6 HW e 2 SW).
Cause	13	Tipo dell'interruzione e bit delle interruzioni pendenti
EPC	14	Registro contenente l'indirizzo dell'istruzione che ha causato l'interruzione.

Registro causa:



Registro stato:

Codici operativi

