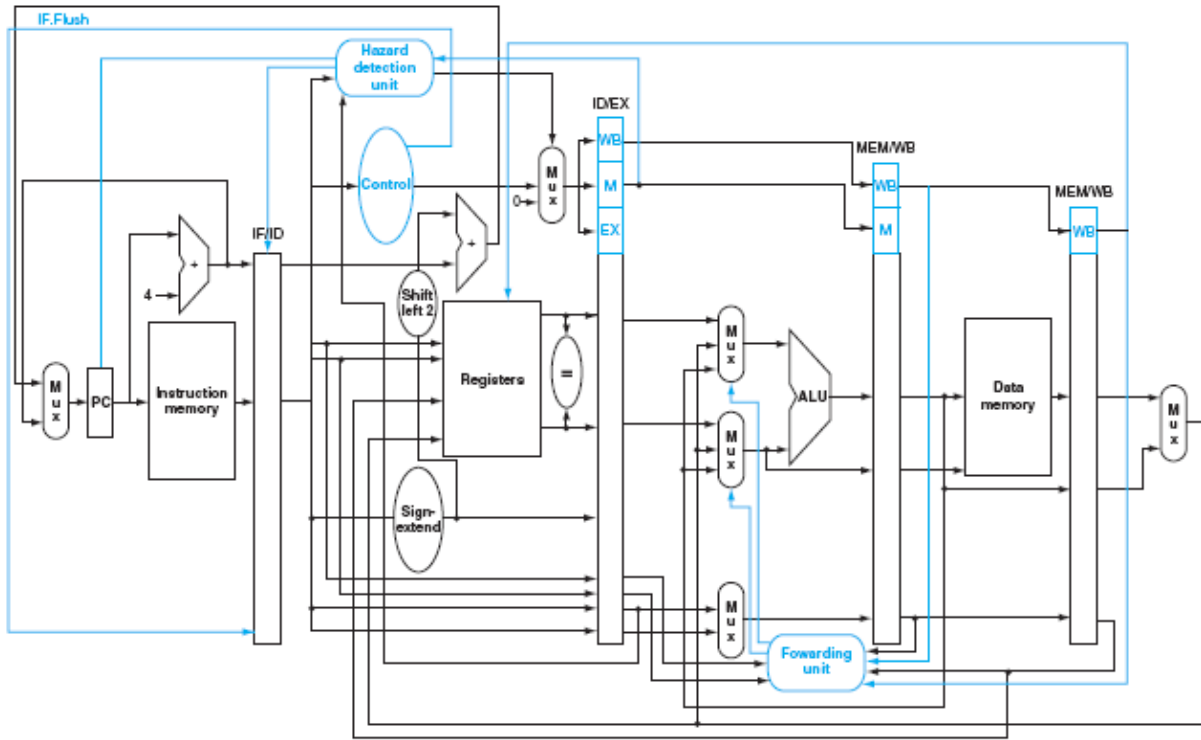


Cognome e nome dello studente:

Matricola:



1. [7] Data la CPU sopra, specificare il contenuto di TUTTE le linee (dati e controllo) quando è in esecuzione il seguente segmento di codice [5]:

```
add $t3, $t2, $t1
or $t4, $t1, $t1
sub $t1, $t2, $t3
lw $t1, 20($s0)
sw $s2, 20($s0)
```

quando l'istruzione di add si trova in fase di WB. Specificare sullo schema (con colore o con tratto grosso) quali linee, all'interno dei diversi stadi, trasportino dati utili all'esecuzione dell'istruzione, riferendosi alla situazione in cui l'istruzione di add è in fase di WB [2].

2. [3] Cos'è un hazard? Cos'è uno stallo? Stallo e bolla sono la stessa cosa? Si verificano hazard nell'esecuzione del codice precedente? Motivare la risposta [3].

3. [6] Cosa sono gli interrupt e le eccezioni? Come vengono gestite dai sistemi operativi? Specificare gli elementi della CPU che sono dedicati alla gestione delle eccezioni e supportano il sistema operativo nel MIPS. Modificare la CPU sopra per potere gestire un'eccezione di "Overflow". Cosa si intende per mascheramento degli interrupt? Viene praticato nei MIPS?

4. [4] Cosa si intende per Superpipeline e pipeline multiple issue? Cosa sono gli "issue"? Descrivere come funziona una pipeline multiple-issue statica e dinamica. Cosa si intende per: a) "Reservation station", "Register renaming", "Commit unit", "Speculazione", "Flush"? Spiegare i motivi che hanno spinto lo sviluppo delle pipeline in queste direzioni.

5. [8] Disegnare una memoria cache (parte dati + TAG + v) per un'architettura MIPS, a 4 vie di 2KByte per banco, e linee di 8 parole (per ciascun banco). Supponiamo che la parola sia costituita da 4 byte (architettura a 32 bit). Supponiamo anche che l'indirizzamento della memoria principale sia su 32 bit. Definire cosa rappresenta il campo TAG e dimensionarlo opportunamente. Supponiamo che all'inizio i bit di validità siano

tutti a 0. Definire cosa succede in corrispondenza di questo frammento di codice (se si verifica una miss, una hit e dove vengono scritti / letti i dati della cache, quale indirizzo e quale tag vengono associati ad ogni istruzione):

```
sw $t0, 24($zero)
lw $t0, 64($zero)
lw $t0, 56($zero)
sw $t0, 56($zero)
sw $t0, 0($zero)
lw $t0, 24($zero)
lw $t0, 64($zero)
lw $t0, 128($zero)
lw $t0, 1024($zero)
lw $t0, 2048($zero)
lw $t0, 2072($zero)
```

6. [4] Perché sono state introdotte le gerarchie di memoria? Cos'è un hit e miss della memoria? Cosa comportano? E' possibile per una memoria cache ad n-vie avere lo stesso TAG su linee diverse? E TAG uguali sulla stessa linea, su banchi diversi? E' possibile per una memoria cache completamente associativa avere lo stesso TAG su linee diverse? Motivare le risposte.

Cosa si intende per write through e write back? Quali vantaggi presenta ciascuna delle due modalità? Cosa si intende per coerenza (di una memoria)? Come si può garantire la coerenza della memoria? Quali sono le tecniche utilizzate? Quali sono i problemi? Cosa si intende per memoria associativa?

7. [2] Cosa si intende per parallelismo implicito ed esplicito? Cos'è un cluster? Cos'è un'architettura multi-core? Quali sono le maggiori problematiche per cluster e architetture multi-core?

8 [2] Cosa si intende per arbitraggio? Cosa si arbitra e chi arbitra? Come avviene una transazione su un bus sincrono e su un bus asincrono? Quali sono i segnali importanti e perchè?

Registri del 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)

