

Cognome e nome dello studente:

Matricola:

1. [5] Data la CPU di Figura 1, specificare il contenuto di **tutti** i registri (parte master e parte slave), quando è in esecuzione il seguente segmento di codice [5]:

```
0x00000400 and $s5, $t2, $t1
0x00000404 sw $t1, 64($s0)
0x00000408 add $t4, $s5, $t1
0x0000040C addi $t1, $s1, 100
0x00000410 lw $s2, 32($t4)
```

quando l'istruzione di `and` 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 [2]. Ci sono hazard nel codice precedente? Motivare la risposta [1].

2. [3] Spiegare il funzionamento della pipeline dell'Intel i7 (Figura 2) identificando le diverse fasi del ciclo di esecuzione. Identificare quali componenti possono essere considerati un'evoluzione di quelli nella pipeline di Figura 1 e perché.

3. [5] Descrivere come funzionano le seguenti tecniche/componenti e discutere se le tecniche/componenti se siano principalmente **software** o **hardware** e **perché**. In alcuni casi la risposta corretta può essere entrambi gli approcci. Identificare quali sono i **punti forti** ed i **punti deboli**.

1. Predizione dei salti
2. Branch prediction buffer
3. Speculazione
4. Parallelizzazione dell'esecuzione
5. Pipeline superscalari
6. Pipeline dotate di VLIW
7. Parallelizzazione a livello di parola
8. Esecuzione fuori ordine
9. Reservation station
10. Buffer di riordino
11. Ridenominazione dei registri
12. Issue
13. Polling
14. DMA
15. Bridge
16. Porta I/O

4. [1] Spiegare chiaramente cosa si intende per stallo e illustrare almeno una tecnica per ottenerlo.

5. [4] Disegnare una memoria cache (parte dati + TAG + bit di validità) e la sua porta di lettura. Utilizzare componenti three-state per minimizzare il numero di porte logiche. Considerate un'architettura MIPS a 32 bit, a 4 vie di 64 KByte per banco, e linee di 8 parole (per ciascun banco). Definire cosa rappresenta il campo TAG e dimensionarlo opportunamente. Dove posso trovare il dato letto dall'istruzione `lw $t1, 2048($0)`? Da quanti bit è costituita questa memoria complessivamente? Cosa succede quando si verifica una miss? Come si può limitare la frequenza di miss? Spiegare cosa si intende per tecnica "Least Recently Used" esatta e approssimata e come viene implementata.

6. [4] Cosa sono gli interrupt e le eccezioni? Come vengono gestiti dalle architetture Intel e dalle architetture MIPS/ARM? Specificare gli elementi della CPU MIPS che sono dedicati alla gestione delle eccezioni e cosa contengono. Modificare la CPU di Figura 1 per potere gestire un'eccezione di "Overflow". Cosa si intende per mascheramento degli interrupt? Viene praticato nei MIPS? Come vengono gestite le eccezioni e gli interrupt dai sistemi operativi sul MIPS?

7. [2] Che cos'è e che cosa rappresenta il "roof model"? Cosa rappresenta l'intensità aritmetica? Si riferisce a una CPU o ad un particolare programma? Descrivere un metodo con il quale si può determinare il roof model. Quali sono i passi per ottimizzare le prestazioni del codice suggeriti dal roof-model? Cos'è un kernel benchmark? Cos'è lo SPEC?

8. [1] Riportare alcune caratteristiche della architetture x86. A cosa serve il post-byte?

9. [3] Cosa si intende per gerarchia delle memorie? Spiegare chiaramente cosa si intenda per **coerenza** e **consistenza** di una memoria. Fare degli esempi. A quali memorie si applicano? Spiegare chiaramente come funzionano i seguenti protocolli che mirano a garantire la coerenza o la consistenza:

- a) Write-back
- b) Write-through
- c) Write invalidate

Cos'è il lock? A cosa serve?

10. [2] Cos'è la memoria virtuale? Cos'è la Tabella delle pagine? Dove si trova? Cos'è il "Translation Lookaside buffer"? Dove si trova? A cosa servono la memoria virtuale, il TLB e la tabella delle pagine? Che relazione c'è tra la memoria virtuale e la memoria fisica? Chi utilizza la memoria virtuale? Chi utilizza la memoria fisica? Cosa succede quando la CPU chiede una parola alla memoria?

11. [3] Costruire una cella di DRAM e una cella di SRAM e discutere le differenze. Cosa si intende per modalità di lettura a burst? Come deve essere organizzata una memoria DRAM per ottenere il massimo dell'efficienza dei trasferimenti verso una cache con linee di 8 parole da 4 Byte?

12. [2] Cosa sono i codici di rilevamento e correzione degli errori? Come funziona il codice di Hamming? Calcolare il codice per il numero binario 1111 1111. Chi utilizza il codice di rilevamento e correzione degli errori? Schizzare uno schema a blocchi di un modulo, che mediante il codice di Hamming, possa correggere un errore singolo.

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)

Figura 1

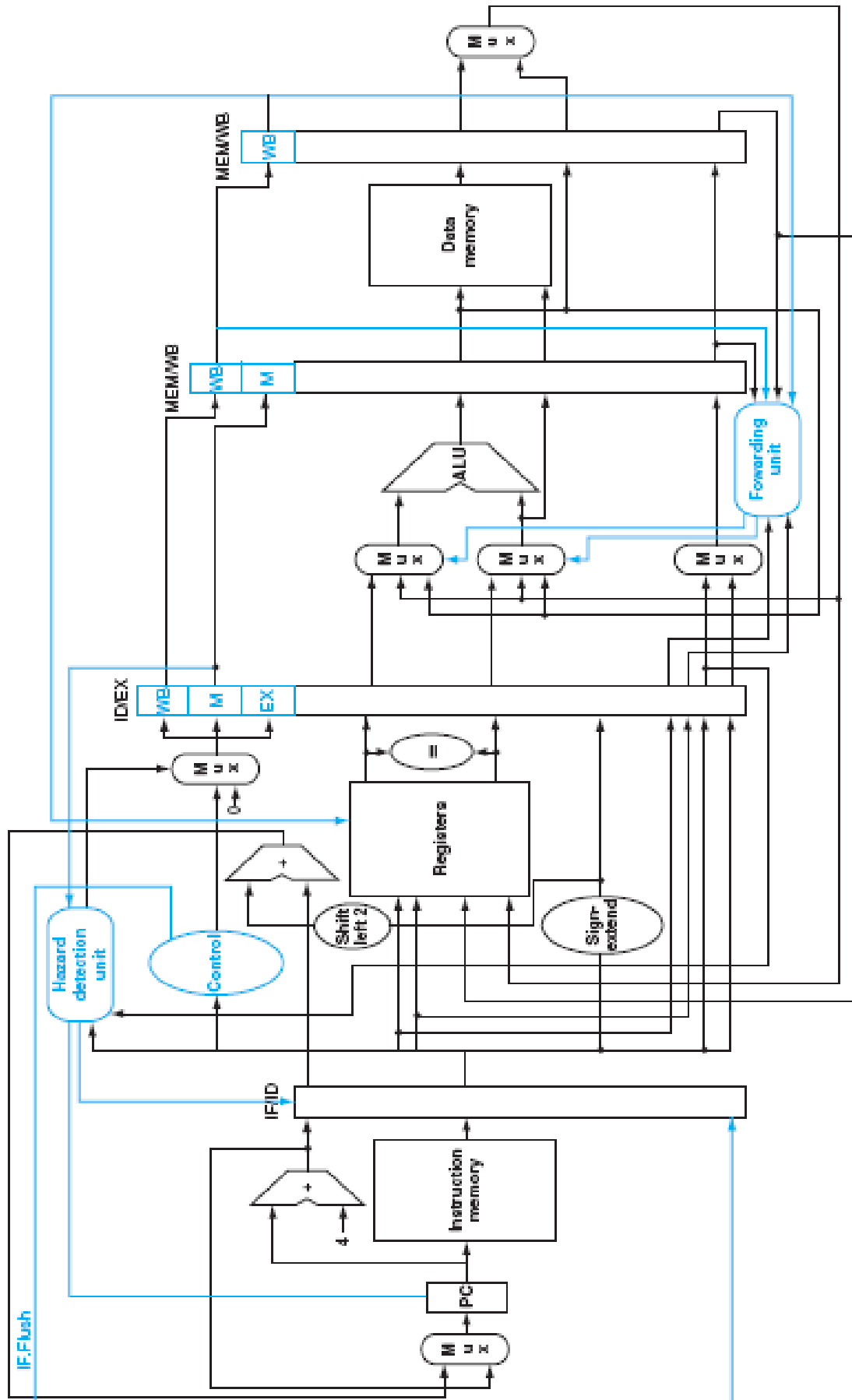


Figura 2

