Esercitazione di ricapitolazione – Architettura modulo II

1. Quando conviene una CPU con pipeline rispetto ad una CPU a ciclo singolo?

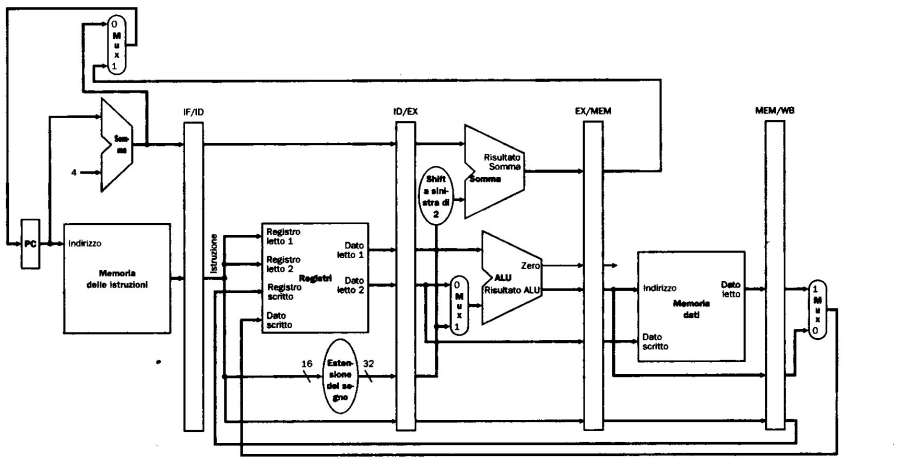
2. Cosa è una CPU con pipe-line? Una pipe-line consente l'esecuzione più veloce di un'istruzione rispetto ad una CPU a singolo ciclo? Di quanto aumenta la velocità di esecuzione in una CPU con pipeline? Una CPU con pipeline richiede più o meno unità funzionali di una CPU a ciclo singolo? Motivare le risposte.

3. Modificare la CPU a singolo ciclo in modo tale che diventi compatibile all'esecuzione in pipe-line (senza gestione di hazard). Modificare la CPU in due modi diversi: tenendo conto dei segnali di controllo e non tenendone conto. Quali diventano i registri? Cosa contengono? Quali passi di esecuzione separano? Da quanti bit sarà costituito ciascun registro? Cos'è uno stallo?

5. Cosa rappresenta un hazard? Quando si verifica? Fare un esempio per ogni tipo di hazard.

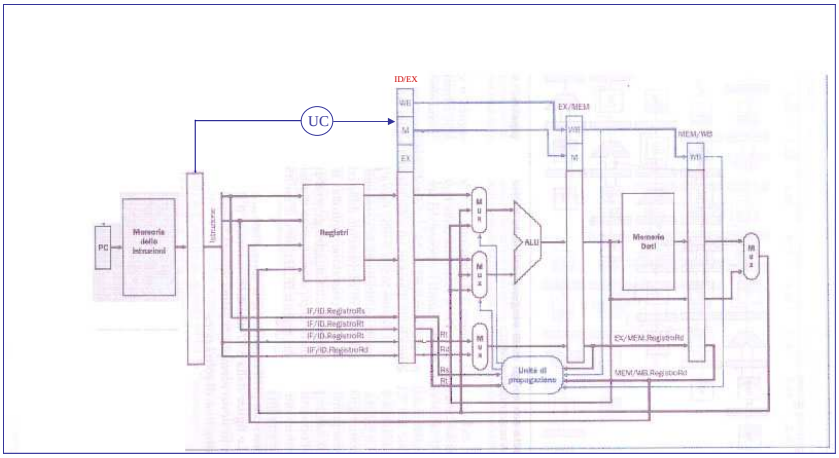
6. Visualizzare con uno schema temporale e con un esempio, quali sono le dipendenze tra le istruzioni che provocano un hazard sui dati o un hazard sul controllo.

7. Dato lo schema a fianco, quale sarà il contenuto dei registri di pipeline (stato), quali saranno i segnali di controllo attivi e quali indifferenti, al termine (prima della commutazione del clock) del terzo stadio dell'istruzione in bold (sub):
add $t0, $t1, $t2
sub $t3, $t3, $t5
beq $t6, $t0, 16
add $t0, $t1, $t3
sapendo che $t0 = 7, i codici operativi di add e sub = 0, beq = 4; il codice Funct della add è 32 e della sub è 34.



8. Modificare lo schema della CPU con pipeline riportato sopra per potere gestire: un hazard sui dati dovuto ad istruzioni aritmetico-logiche. Scrivere le condizioni logiche che vengono utilizzate per identificare questo hazard e le funzioni logiche che servono a risolverlo.

9. Modificare lo schema della CPU, in modo tale da potere gestire:
a) Un hazard dovuto ad un'istruzione di load.
b) Un hazard sul controllo dovuto ad una beq.
Dare un esempio di codice in cui questi due hazard si verificano. E spiegare la



dipendenza tra dati che origina l'hazard.

10. Dato lo schema qui a fianco, specificare il contenuto di tutti i registri ed i segnali di controllo negli stadi 1, 2, 3, 4, 5 di esecuzione della lw:
add $t0, $t1, $t2
addi $t0, $t1, 64
beq $t3, $t4, 16
lw $t3, 0($t0)
add $t6, $t6, $t7
add $t4, $t5, $t3
sapendo che i codici operativi di add, addi, beq, lw sono rispettivamente: 0, 8, 4, 35. Il codice funct della add è 32 e che $t0 = $7.

11. Modificare la CPU disegnata sopra in modo tale da potere gestire gli hazard generati da una lw. Specificare tutti i segnali di controllo.

12. Modificare la CPU disegnata sopra in modo tale da potere gestire gli hazard generati da una branch. Specificare tutti i segnali di controllo.

13. A cosa servono i registri causa e stato? Cosa si intende per mascheramento di un interrupt?

14. Descrivere tutte le operazioni necessarie alla gestione di un interrupt o eccezione.

15. Modificare la CPU di cui sopra, perchè sia in grado di gestire le eccezioni di overflow e di codice operativo non valido. Cosa è un'eccezione e un'interruzione? Che ruolo hanno i registri EPC, il registro Causa, Stato? Cos'è la maschera di interruzione? Quali sono le due modalità di risposta ad un'interruzione o eccezione? Cosa è il coprocessore 0? Come viene gestito? Cos'è un exception handler? Quali sono i passi da eseguire per servire in modo corretto un'eccezione? Quali sono i passi da eseguire per servire in modo corretto un interrupt? Ci sono differenze?

16. Definire i tipi di istruzioni disponibili. Quali di queste frasi sono corrette?
1) Tutte le istruzioni aritmetico-logiche sono di tipo R.
2) Le istruzioni di accesso alla memoria sono di tipo I.
3) Le istruzioni di salto sono di tipo I.
4) Le istruzioni con due operandi sorgenti su 32 bit sono di tipo R.
5) Le istruzioni di tipo R sono tutte istruzioni aritmetico-logiche.
6) Le istruzioni di tipo R hanno 2 operandi sorgenti su 32 bit.

17. Cos'è il branch prediction buffer? Disegnare una macchina a stati finiti che cambi la predizione su un salto **dopo 2 errori**. Perchè di solito si utilizza un branch prediction buffer con due bit per la predizione del salto? Cosa si intende per esecuzione speculativa? Come si può migliorare il funzionamento di una pipeline per la gestione dei salti condizionati?

18. Cosa si intende per pipeline superscalare, superpipeline e scheduling dinamico? Che cosa sono le reorder station? E le reservation station? A cosa serve l'operazione di renaming dei registri? E' corretto dire che internamente la CPU di un Pentium IV ha un'ISA RISC e perchè? Quante cache primarie ha un Pentium IV e perchè? Cosa si intende per esecuzione fuori-ordine?

18. Che differenza c'e' tra un multi-core e un cluster? Cos'è lo snooping? Chi lo attua? Perchè? Cos'è la coerenza? Come può essere garantita dall'hardware?

19. Definire lo schema di massima dell'Architettura della pipe-line di un Pentium IV e descrivere quali sono i componenti principali e quali possono essere i problemi principali.

20. Qual'è il ruolo della memoria? Quali funzioni si possono eseguire sulla memoria? Cosa rappresenta l'altezza e l'ampiezza della memoria e come si calcola la capacità? Come è definita la parola di memoria? Qual'è la relazione tra capacità di memoria e numero di bit di indirizzamento?

21. Cosa esprime il principio di località di una memoria? Cosa contiene una memoria cache? Può il contenuto di una memoria cache essere diverso dal contenuto della memoria principale?

22. Cosa si intende per Hit e Miss? Hit rate e miss rate? Può essere la somma di Miss rate e Hit rate maggiore di 15? Cosa si intende per Write through e Write back in una memoria cache? Cosa si intende per LRU? Cosa si intende per modalità a buffer circolare?

23. Data una memoria cache di 64Kbyte ed una RAM di 1Gbyte, a quanti bit devo dimensionare il campo TAG in caso di una memoria cache a mappatura diretta? E in caso di una cache a 2 vie?

24. Dato un indirizzo di memoria di 32 bit, specificare il significato dei singoli bit nel caso di utilizzo di una memoria cache con le seguenti caratteristiche:

Cache a mappatura diretta di 128Kbyte, con linee contenenti 8 parole di 4 byte ciascuna.
Cache a 2 vie di 128Kbyte, con 2 banchi e linee contenenti 8 parole di 4 byte ciascuna.
Cache associativa di 128Kbyte con linee ciascuna contenente 8 parole di 4 byte ciascuna.
Per ciascuna delle 3 cache disegnare il circuito di lettura e scrittura.

25. Disegnare le seguenti tre memorie cache:
Cache a mappatura diretta di 128 byte con linee contenenti 2 parole di 4 byte.
Cache a 2 vie di 128 byte con linee contenenti 2 parole di 4 byte.
Cache associativa con linee contenenti 2 parole di 4 byte.
Data l'istruzione lw $t0, 1024($zero), specificare all'interno delle cache a), b), c) dove si trova la parola che deve essere letta?

26. Specificare per una memoria cache a chi viene inviato: il segnale di Miss, il segnale di Hit, il dato letto; e da dove proviene il dato scritto.

27. Cos'è l'interleaving di una memoria? Come è costruita una memoria SRAM? Cosa significa l'acronimo SRAM? Qual'è il ruolo dell'uscita "three-state" in una memoria SRAM? Discuterlo con un esempio. Come vengono gestiti i banchi di memoria?

28. Qual'è il principio di funzionamento di una DRAM? Cosa è una SDRAM? Quando una SDRAM lavora in "burst mode"? Cosa rappresentano i segnali CAS e RAS di una DRAM? Cos'è il refresh della memoria? Si può leggere la memoria durante il refresh? Perché? Può una SDRAM essere letta in modalità asincrona?

29. Qual è il vantaggio / svantaggio dell'organizzazione a matrice di una memoria?

30. Dato il seguente segmento di codice, descrivere istruzione per istruzione cosa succede in una cache a 2 vie, dove ciascun banco è di 1Kbyte con linee di 4 word, e con tutte le linee con dati non validi:
lw $s0, 64($zero)
lw $s0, 8($zero)
lw $s0, 1032($zero)
lw $s0, 2056($zero)
lw $s0, 4($zero)
sw $s1, 4($zero)

31. Disegnare la porta di lettura e scrittura di una cache a mappatura diretta di 2Kbyte e 8 linee, di una cache a 2 vie con banchi di 2Kbyte e linee di 8 word e di una memoria cache associativa di 2Kbyte e 8 linee.

32. Dato un indirizzo di 32 bit, come vengono utilizzati i bit per indirizzare una memoria cache a k-vie?

33. Quali requisiti imporreste alla cache primaria? E alla cache secondaria? Data una dimensione di cache, cosa succede all'aumentare della lunghezza di una linea e all'aumentare del numero di vie?

34. Cosa si intende per split-cache?

35. Come si utilizza la tecnologia three-state all'interno delle memorie? Si applica indifferentemente alle memorie SRAM e DRAM? Perché?

36. Cos'è un bus? Cos'è l'arbitraggio? Cosa rappresenta il segnale di "bus grant"? Descrivere gli schemi di arbitraggio centralizzato e gli schemi di arbitraggio distribuiti (con e senza autoselezione).

37. Descrivere i requisiti di funzionamento dei tre tipi di bus principali: processore-memoria, backplane e I/O. Come viene sincronizzata la trasmissione di dati sul bus?

38. Cosa è il device controller? Quali sono i suoi componenti principali? A cosa serve? Come vengono indirizzate le periferiche?

39. Descrivere una procedura di handshaking tipica per l'accesso ad un bus sincrono ed asincrono. Cosa si intende per bus-master? A cosa si riferisce: al device controller, al bridge o al dispositivo?

40. Cosa si intende per transazione sul bus?

41. Come viene gestito l'I/O dall'ISA di un'architettura MIPS e da un'architettura INTEL?

42. Come viene gestito l'I/O a controllo di programma? Cos'è il polling? Come viene gestito l'I/O tramite interrupt? Cosa è il DMA? Che cos'è lo "spin lock"?

43. Esercizio. Supponiamo di valutare il costo per una CPU con frequenza pari a 1Ghz per trasferire 64Mbyte di dati da un *Hard-disk. Supponiamo che ad ogni accesso vengano trasferiti 64 byte. Il tempo richiesto ad una CPU per il trasferimento di 64 byte è di 200 cicli_clock che devono essere sommati al tempo per accedere alla periferica che è di 300 cicli di clock in modalità a controllo di programma e di 400 cicli di clock in modalità interrupt. Confrontate le prestazioni con un trasferimento in DMA nel quale vengono*

trasferiti 6,400 parole tenendo conto che il tempo richiesto per l'avviamento della DMA è di 500 cicli di clock e per la chiusura è di 800 cicli di clock. Quale tra le modalità a controllo di programma e ad interrupt è più efficiente in questo caso (motivare la risposta)?
Valutiamo le diverse modalità di trasferimento:
a controllo di programma (300 cicli di clock)
ad interrupt (400 cicli di clock)
mediante DMA (500 cicli clock per l'avviamento e 800 cicli di clock per la chiusura).
A controllo di programma: 64Mbyte / 64byte = 1M accessi. Tempo di CPU: 1Maccessi * (200+300)cicli = 500 * 10^6. Percentuale di sfruttamento della CPU: 50%.
Mediante interrupt: 64Mbyte / 64 byte = 1M interrupt. Tempo di CPU: 1Minterrupt * (200+400) cicli di clock. Percentuale di utilizzo della CPU: 60%.
Mediante DMA: 64Mbyte / 6400 word = 2,500 DMA => 2,500 * (500 + 800) = 3,25 Mega cicli di clock. Percentuale di utilizzo della CPU: 0,325%.
Ripetete le valutazioni per trasferire dati da tastiera (i quali richiedono una frequenza di 10byte/s, un tempo di CPU di 10 cicli_clock / byte per l'operazione di I/O e trasferisce 1 byte per ogni accesso).
A controllo di programma: 10 accessi / s * (10 + 300) cicli_clock = 3100 cicli_clock. Percentuale di sfruttamento della CPU: 3,1 / 10^6.
Mediante interrupt: 10 accessi / s * (10 + 400) cicli_clock = 4100 cicli_clock. Percentuale di utilizzo della CPU: 4,1 / 10^6.
Mediante DMA. Non ha senso. Trasferisco 1 byte alla volta.

44. Descrivere l'utilizzo principali dei registri nelle architetture IA-32. Cosa sono i segmenti? Modalità di indirizzamento dei dati nelle architetture IA-32. Come vengono gestite le operazioni di I/O nelle IA-32? Come funzionano i modificatori delle istruzioni dell'ISA delle architetture IA-32?

45. Cosa è lo SPEC? Cosa è il CPI? Perchè sono nati i benchmark? Enunciare la legge di Amdhal. Sotto quali ipotesi riesco ad aumentare la velocità di esecuzione in architetture multi-processori? E la velocità di un server?

46. Si deve valutare un miglioramento di una macchina per l'aggiunta di una modalità vettoriale. La computazione vettoriale è 20 volte più veloce di quella normale. La *percentuale di vettorizzazione* è la porzione del tempo che può essere spesa usando la modalità vettoriale.
Disegnare un grafico che riporti lo speedup come percentuale della computazione effettuata in modo vettoriale.
Quale percentuale di vettorizzazione è necessaria per uno speedup di 2?
Quale percentuale di vettorializzazione è necessaria per raggiungere la metà dello speedup massimo?
La percentuale di vettorizzazione misurata è del 70%. I progettisti hardware affermano di potere raddoppiare la velocità della parte vettoriale se vengono effettuati significativi investimenti. Il gruppo che si occupa dei compilatori può incrementare la percentuale d'uso della modalità vettoriale.
Quale incremento della percentuale di vettorizzazione sarebbe necessario per ottenere lo stesso guadagno di prestazioni?
Quale investimento raccomandereste?

47. Descrivere l'architettura dei bus di un'architettura Intel (Pentium 4). Quale funzione svolge ciascun bus?

48. Dove si trova fisicamente il circuito di arbitraggio di un bus? E' un circuito sequenziale o combinatorio (motivare la risposta)?

49. Quali circuiti vengono utilizzati per un SRAM: flip-flop o latch? Motivare la risposta.

50. Descrivere la struttura fisica interna di un disco. Fare uno schizzo ed indicare quali sono i settori e le tracce.

51. Cosa si intende per multiple zoned recording? Perché si utilizza?

52. Cosa rappresenta il tempo di seek? E' costante all'interno di uno stesso disco? E' costante per tutto i dischi?

53. Cosa rappresenta il tempo di attesa? Può essere uguale su dischi diversi e perché?

54. Cosa si intende per gerarchia di memoria? Perché viene richiesta una gerarchia?

55. Cosa si intende per bus seriale? Che differenza c'è tra bus PCI-Express e PCI64?

56. Definire il ruolo dei bridge nella architetture? Dove si trovano? Cosa prevedete ci sia all'interno di un bridge?

57. Modalità di accesso ai dati nella memoria principale (RAM) nelle architetture MIPS [2].

58. Cosa succede in un'operazione di push o di pop? Quale parte dell'architettura interessa? [1]

59. Cosa rappresenta il "roof model"? Cosa rappresenta l'intensità aritmetica? Si ricerisce ad una CPU o ad un particolare programma? Cos'e' lo SPEC? Cosa misura 1 FLOPS? Quali sono i passi suggeriti per ottimizzare il codice su un'architettura parallela?

60. Cos'e' un cluster? Quali sono i problemi maggiori legari ai cluster e ai multi-core?

61. Qual'è la differenza tra parallelismo esplicito ed implicito?

62. Data una CPU quad-core, a 32 bit con 4 cammini di calcolo, in ciascuno dei quali vengono elaborati dati a 128 bit. Detta CPU ha un clock di 2GHz ed invia ad esecuzione 2 istruzioni per ogni ciclo di clock. A questa CPU è associato un sistema di memoria che è in grado di sostenere l'elaborazione con un flusso dati dalla memoria alla CPU pari a 2Gbyte /s. Determinare la massima velocità di elaborazione della CPU per 6 diversi programmi benchmark che hanno intensità aritmetica rispettivamente di: ½, 1, 4, 8, 16, 32.

63. Dato un programma con il seguente MIX di istruzioni: accesso a memoria (20%), Branch (14%), Operazioni (60%), Jump (6%). Suppondo che i tempi di esecuzione delle istruzioni appartenenti alle quattro diverse classi sia rispettivamente: 10ms, 6ms, 8ms, 2ms definire qual'è l'aumento di perstazioni che si ottiene se:
a) la velocità di esecuzione delle operazioni viene triplicata.
b) la velocità di esecuzione delle branch (tenuto conto delle criticità) viene dimezzata.
c) la velocità di esecuzione delle istruzioni di accesso a memoria viene quadruplicata.
Definire il massimo incremento di prestazioni possibile per un miglioramento dell'esecuzione delle operazioni appartenenti alle singole classi.

64. Dato un programma con il seguente MIX di istruzioni: accesso a memoria (30%), Branch (15%), Operazioni (50%), Jump (5%). Suppondo che i tempi di esecuzione delle istruzioni appartenenti alle quattro diverse classi sia rispettivamente: 10ms, 6ms, 8ms, 2ms definire qual'è l'aumento di perstazioni che si ottiene se:
a) la velocità di esecuzione delle operazioni viene triplicata.
b) la velocità di esecuzione delle branch (tenuto conto delle criticità) viene dimezzata.
c) la velocità di esecuzione delle istruzioni di accesso a memoria viene quadruplicata.
Definire il massimo incremento di prestazioni possibile per un miglioramento dell'esecuzione delle operazioni appartenenti alle singole classi.

65. Cosa si intende per Mflop? Cos'è un kernel benchmark? Quando si può utilizzare? Perché si utilizzano i benchmark? Quali sono i limiti dei benchmark? [1]

66. Definire almeno due metriche di valutazione delle prestazioni.

67. Consider a SEC code that protects 8 bit words with 4 parity bits. If we read the value 0x375, is there an error? If so, correct the error.

68. The following data constitutes a stream of virtual addresses as seen on a system. Assume 4 KByte pages, a 4-entry fully associative TLB, and true LRU replacement. If pages must be brought in from disk, increment the next largest page number. The following stream of virtual addresses is seen on a system: 4669, 2227, 13916, 34587, 48870, 12608, 49225.

Page table

Valid	Physical Page or in Disk
1	5
0	Disk
0	Disk
1	6
1	9
1	11
0	Disk
1	4
0	Disk
0	Disk
1	3
1	12

TLB

Valid	Tag	Physical Page Number
1	11	12
1	7	4
1	3	6
0	4	9

1) Given the address stream shown, and the initial TLB and page table states provided above, show the final state of the system. Also list for each reference if it is a hit in the TLB, a hit in the page table, or a page fault.
2) Use 16 KByte pages instead of 4 Kbyte pages. What would be some of the advantages of having a larger page size? What are some of the disadvantages?
3) Show the final contents of the TLB if it is 2-way set associative. Also show the contents of the TLB if it is direct mapped. Discuss the importance of having a TLB to high performance. How would virtual memory accesses be handled if there were no TLB?