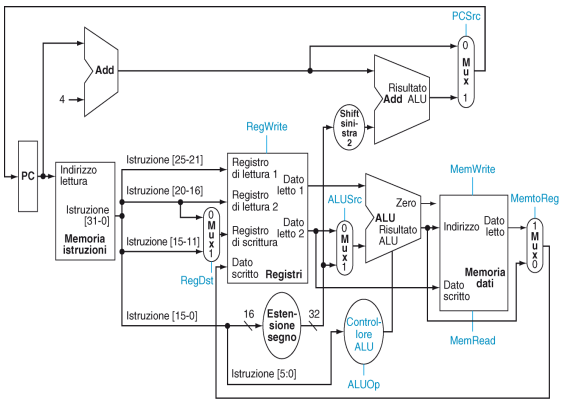

Figure 1: Schema di datapath

Rispondere alle domande a risposta multipla annerendo la casella corrispondente alla risposta corretta. Ogni domanda ha una ed una sola risposta corretta.

Cognome e Nome: .....

Matricola: .....

Domanda 1 Con riferimento alla figura 1, si dica quale delle seguenti affermazioni è corretta.

- ☐ Il blocco addizionatore in alto a destra è usato per implementare i salti condizionati. Il blocco shift a sinistra di due che lo precede è necessario per trasformare l'indirizzo del salto da offset relativo a valore assoluto.
- ☐ Il blocco addizionatore in alto a destra viene utilizzato per calcolare l'indirizzo da cui prelevare un dato richiesto per un'operazione di `lw`.
- ☐ Il blocco addizionatore in alto a destra è usato per implementare i salti condizionati. Il blocco shift a sinistra di due che lo precede è necessario per trasformare l'indicazione dell'indirizzo da word a byte.
- ☐ Nessuna delle altre risposte

Domanda 2 Si consideri una fully associative grande $16KB$, con blocchi di 64 byte per blocco. In che blocco di cache è mappata la parola che sta all'indirizzo $0x100620$?

- ☐ Nel blocco numero 48 o nel blocco numero 49.
- ☐ Nel blocco numero 24.
- ☐ Nel blocco numero 0 o nel blocco numero 1 o nel blocco numero 2 o nel blocco numero 3.
- ☐ Nessuna delle altre risposte.
- ☐ Nel blocco numero 32.

Le cache fully associative hanno il tag memorizzato per intero e non utilizzano gli ultimi bit per decidere la posizione.

Domanda 3 Individua l'espressione logica equivalente a: $F = x \cdot (y + z) + \overline{x + \overline{z}}$

- [ ] Nessuna delle altre risposte
- [ ] $F = z \cdot \overline{z} + \overline{x}$
- [x] $F = x \cdot y + z$
- [ ] $F = x \cdot y + \overline{x} \cdot z$
- [ ] $F = 1$

F = x*(y+z) + neg(x+neg(z))=
= x*(y+z) + neg(x)*neg(neg(z))
= x*y + x*z + neg(x)*z
= x*y+z(x+neg(x)
= x*y+z

Domanda 4 Si consideri una CPU dotata di 2 cache separate per dati ed istruzioni. Il CPI ideale della CPU è 4, la cache istruzioni ha una frequenza di miss del 1% e la cache dati ha una frequenza di miss del 4%. Supponendo che un cache miss richieda 100 cicli di clock per essere servito e che il 20% delle istruzioni Assembly accedano a dati in memoria, il CPI reale (il numero di cicli necessari in media per eseguire un'istruzione tenendo conto degli stalli per accesso alla RAM) è:

- [ ] 3.72
- [x] Nessuna delle altre risposte
- [ ] 7.44
- [ ] 3.44
- [ ] 8

Cache → Dati → Freq. miss 4%
Cache → Istruzioni → Freq. miss 1%

CPI ideale = 4
1 cache miss = 100 cicli di clock
20% accedono in memoria

CPI reale = CPI ideale + Penaliz. Istruz. + Penaliz. Dati =
4+0.01*100+0.04*100*0.2

A B C B

A: Freq. miss istruzioni
B: Penalità miss
C: Freq. miss dati

Domanda 5 Usando la rappresentazione binaria, svolgere la somma $183 + 37$

- [ ] $183_{10} + 37_{10} = 101011100_2$
- [ ] $183_{10} + 37_{10} = 11011001_2$
- [x] $183_{10} + 37_{10} = 11011100_2$
- [ ] $183_{10} + 37_{10} = 11001101_2$
- [ ] Nessuna delle altre risposte

Domanda 6 Inizialmente il contenuto di `s0` = 0000 0000 0000 0000 0000 0000 0000 0000 Quale valore conterrá `s0` dopo aver eseguito le seguenti istruzioni Assembly?

```
addi $t0, $zero, 0x000F
sll $t0, $t0, 28
or $s0, $s0, $t0
```

- [x] `s0` = 1111 0000 0000 0000 0000 0000 0000 0000
- [ ] Nessuna delle altre risposte
- [ ] `s0` = 1111 1111 0000 0000 0000 0000 0000 0000
- [ ] `s0` = 1111 0000 0000 0000 0000 0000 0000 1111
- [ ] `s0` = 0000 1111 0000 0000 0000 0000 0000 1111

Domanda 7 I *simboli definiti* contenuti in un file oggetto (`.o`) non eseguibile:

- [ ] Non sono associati ad indirizzi di memoria ne' assoluti ne' relativi; l'associazione ad un indirizzo di memoria avverrà solo nella fase di linking
- [x] Sono associati ad indirizzi di memoria *relativi*
- [ ] Nessuna delle altre risposte
- [ ] Nei file `.o` non esiste alcuna nozione di "simboli definiti"
- [ ] Sono associati ad indirizzi di memoria *assoluti*

ToolChain - Slide 9

Domanda 8 L'istruzione `lea (%rdi, %rsi), %rax`:

- [ ] Carica in `%rax` il contenuto della locazione di memoria indicata da `%rdi` + `%rsi`
- [ ] Nessuna delle altre risposte
- [x] Somma il contenuto dei registri `%rdi` ed `%rsi` salvando il risultato in `%rax`
- [ ] Non è un'istruzione Assembly valida
- [ ] Salva il contenuto di `%rax` nella locazione di memoria indicata da `%rdi` + `%rsi`

Domanda 9 Si consideri una CPU in cui le 5 fasi di esecuzione di un'istruzione impiegano $100ps$, $400ps$, $600ps$, $300ps$ e $100ps$. L'incremento di prestazioni che ci si può attendere usando una pipeline è:

- [ ] di 2 volte
- [ ] di 3 volte
- [ ] Nessuna delle altre risposte
- [ ] di 3.5 volte
- [x] di 2.5 volte

Domanda 10 Si consideri la seguente istruzione: `movb` (%rsi, %rax), %bl . Si dica quale delle seguenti alternative è vera:

- [ ] E' un'istruzione ARM sbagliata sintatticamente
- [ ] Nessuna delle altre risposte
- [ ] E' un'istruzione MIPS sbagliata sintatticamente
- [x] E' un'istruzione INTEL che sposta un byte dall'indirizzo di memoria puntato da %rsi + %rax nel sotto-registro %bl
- [ ] E' un'istruzione INTEL che sposta nel registro %bl la word a 64bit puntata da %rsi+%rax

Domanda 11 Nell'architettura MIPS, un gestore di eccezione:

Non l'abbiamo fatto :(

- [ ] E' pre-programmato in un "coprocessore" e non può essere modificato dall'utente.
- [ ] Nessuna delle altre risposte
- [ ] Viene eseguito dalla CPU solo dopo che tutte le fasi di polling sono terminate.
- [x] Può leggere in registri speciali (`Cause`, `EPC` ed altri) il motivo dell'eccezione, l'indirizzo della prossima istruzione da fetchare dopo aver gestito l'eccezione, etc...
- [ ] Non è in grado di capire la causa di un'eccezione.

Questa è quella più lunga, quindi dovrebbe essere giusta

Domanda 12 I quattro bit di controllo della ALU sono generati da:

- [x] :) un'unità di controllo che riceve in ingresso il campo *funct* prelevato dall'istruzione e i due bit detti ALUop
- [ ] un'unità di controllo che riceve in ingresso il campo *funct* prelevato dall'istruzione
- [ ] un'unità di controllo che riceve in ingresso i campi *funct* e *shamt* prelevati dall'istruzione e i due bit detti ALUop
- [ ] Nessuna delle altre risposte
- [ ] un'unità di controllo che riceve in ingresso i due bit detti ALUop