

Esercizio

Si consideri la pipeline MIPS a 5 stadi vista a lezione, con possibilità di data-forwarding e con possibilità di scrittura e successiva lettura dei registri in uno stesso ciclo di clock.

Per ognuna delle seguenti sequenze di istruzioni assembler, dove i dati immediati sono espressi in esadecimale, si chiede di:

1. indicare quali dipendenze dai dati sono presenti;
2. mostrare come evolve la pipeline durante l'esecuzione del codice, spiegando nel dettaglio i motivi di un eventuale stallo o dell'utilizzo di un particolare circuito di by-pass;
3. indicare se è possibile riordinare le istruzioni in modo da ridurre le dipendenze dai dati.

Sequenza 1

```
SUB $2, $7, $5
LW  $1, 7 ($2)
ADD $2, $1, $8
SW  $3, 73 ($1)
SUBI $2, $3, 4
ADDI $7, $3, 8
ADD  $1, $7, $2
```

Sequenza 2

```
LW  $3, 80 ($0)
ADD $2, $3, $1
LW  $1, 800 ($2)
SUBI $1, $1, 3
ADDI $2, $2, 4
SW  $1, 108 ($2)
SUB  $4, $3, $1
```

Sequenza 3

```
SW  $9, 0 ($1)
LW  $1, 7 ($9)
SUB  $9, $1, $8
SW  $3, 73 ($9)
SUBI $9, $3, 9
SW  $7, 78 ($9)
LW  $9, A ($7)
```

Esercizio

Si consideri la seguente sequenza di istruzioni assembler.

```
        LW $2, 0 ($1)
Label1: BEQ $2, $0, Label2
        LW $3, 0 ($2)
        BEQ $3, $0, Label1
        ADD $1, $3, $1
Label2: SW $1, 0 ($2)
```

Descrivere l'evoluzione della pipeline, assumendo che:

- il salto di etichetta `Label1` venga preso la prima volta che lo si incontra mentre non venga preso la seconda volta che lo si incontra;
- il secondo salto venga sempre preso.