

## esercizio pipeline con data-forwarding

Sia data la seguente sequenza di istruzioni assembler, dove i dati immediati sono espressi in esadecimale

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

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:

- 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.

### Soluzione

|                   | 1  | 2  | 3  | 4   | 5   | 6   | 7  | 8   | 9   | 10  | 11  | 12 | 13 | 14 | 15 |
|-------------------|----|----|----|-----|-----|-----|----|-----|-----|-----|-----|----|----|----|----|
| SUB \$2, \$7, \$5 | IF | ID | EX | MEM | WB  |     |    |     |     |     |     |    |    |    |    |
| LW \$1, 7(\$2)    |    | IF | ID | EX  | MEM | WB  |    |     |     |     |     |    |    |    |    |
| ADD \$2, \$1, \$8 |    |    | IF | ID  | EX  | MEM | WB |     |     |     |     |    |    |    |    |
| SW \$3, 73(\$1)   |    |    |    | IF  | IF  | ID  | EX | MEM | WB  |     |     |    |    |    |    |
| SUBI \$2, \$3, 4  |    |    |    |     |     | IF  | ID | EX  | MEM | WB  |     |    |    |    |    |
| ADDI \$7, \$3, 8  |    |    |    |     |     |     | IF | ID  | EX  | MEM | WB  |    |    |    |    |
| ADD \$1, \$7, \$2 |    |    |    |     |     |     |    | IF  | ID  | EX  | MEM | WB |    |    |    |

EX.ALUOutput\_sub -> EX.Top\_ALU\_input\_lw  
 MEM.LMD\_lw -> EX.Top\_ALU\_input\_add  
 MEM.ALUOutput\_subi -> EX.Bottom\_ALU\_input\_add  
 EX.ALUOutput\_addi -> EX.Top\_ALU\_input\_add