

Esercizio

es8

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

LW \$1, 0(\$2)
 ADD \$2, \$3, \$1
 SW \$2, 21(\$1)
 BEQ \$2, \$1, 11
 ADD \$3, \$2, \$2
 ADD \$1, \$1, \$3
 SW \$3, 0(\$1)

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:

- assumendo che la condizione di salto sia falsa, 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 (da compilare)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
LW \$1, 0(\$2)	IF	ID	EX	MEM	WB															
ADD \$2, \$3, \$1		IF	ID	ID	EX	MEM	WB													
SW \$2, 21(\$1)			IF	IF	ID	ID	ID	EX	MEM	WB										
✓ BEQ \$2, \$1, 11					IF	IF	IF	ID	EX	MEM	WB									
ADD \$3, \$2, \$2								IF	ID	EX	MEM	WB								
ADD \$1, \$1, \$3									IF	ID	EX	MEM	WB							
SW \$3, 0(\$1)										IF	ID	ID	EX	MEM	WB					

Commenti alla soluzione:

Tra LW e ADD c'è un RAW ~~\$1~~ e dal momento che \$1 dipende da \$1 in un registro effettuo uno stallo e forward da MEM: MEM.LRD \rightarrow ~~EX~~ Top ALUInput-add ✓
 SW nella riga 3 necessita di \$2 ma questo viene rilasciato in WB (prima parte) di ADD quindi devo avere per NOP. ✓ + RAW \$1 di ✓

Tra ADD e ADD c'è un RAW tra \$3 ed essendo due operazioni aritmetiche faccio il forwarding da EX: ~~EX~~ EX-ALUoutput-add \rightarrow EX-~~EX~~ Top ALUinput-add ✓

Tra SW ci sono 2 dipendenze la 1° con l'ADD della riga 5 che tratta \$3 fino al suo WB, poi quindi aggiungo uno stallo, poi \$1 con l'ultimo ADD, ~~adesso~~ ho una RAW, e dal momento che ho superato la fase EX faccio il forwarding da MEM: MEM.ALUoutput-add \rightarrow EX-Top ALUInput-add ✓
 ✓ put-sw