

Esercizio 1

Progettare il Sottosistema di Calcolo (SCA) di un processore RISC, con organizzazione **pipeline**, con il minor numero di stadi, atto a supportare l'esecuzione delle istruzioni di salto condizionato e logico/aritmetiche di seguito riportate:

sintassi

semantica

jumpX indirizzo

(se flag X = 1 allora PC=PC+4+indirizzo)

sub \$regdest, \$regsorg, indirizzo

(\$regdest = \$regsorg - MEM[indirizzo])

add \$regdest, \$regsorg, indirizzo

(\$regdest = \$regsorg + MEM[indirizzo])

and \$regdest, \$regsorg, indirizzo

(\$regdest = \$regsorg **and** MEM[indirizzo])

or \$regdest, \$regsorg, indirizzo

(\$regdest = \$regsorg **or** MEM[indirizzo])

Descrivere il formato delle istruzioni ipotizzando che siano visibili 8 registri, che non ci siano altre istruzioni e che il numero di bit a disposizione sia 32.

Esercizio 2

Modificare l'architettura dell'esercizio precedente per evitare tutti i possibili conflitti.

CODIFICA ISTRUZIONI

JUMP X INDIRIZZO JUMP

SE (FLAGX == 1) ALLORA PC= PC+4+INDIRIZZO JUMP

OPCODE	FLAG	INDIRIZZO JUMP
3	3	26

OR \$DEST, \$SORG, INDIRIZZO

DEST= SORG OR MEM[INDIRIZZO]

SUB \$DEST, \$SORG, INDIRIZZO

DEST= SORG - MEM[INDIRIZZO]

ADD \$DEST, \$SORG, INDIRIZZO

DEST= SORG + MEM[INDIRIZZO]

AND \$DEST, \$SORG, INDIRIZZO

DEST= SORG AND MEM[INDIRIZZO]

OPCODE	SORG	DEST	INDIRIZZO
3	3	3	23



