

P1)

$$A) CPI = \frac{\#ciclos}{\#instr.} = \frac{\text{tiempo CPU} \times \text{tasa reloj}}{\#instr.} \rightarrow \frac{\#instr.}{\text{tiempo CPU}} = \frac{\text{tasa reloj}}{CPI}$$

↓
IPS

$$IPS_{P_1} = 2 \times 10^9 \text{ instr/s}$$

$$IPS_{P_2} = 2,5 \times 10^9 \text{ instr/s}$$

$$IPS_{P_3} = 1,82 \times 10^9 \text{ instr/s}$$

} P_2 tiene mejor rendimiento

$$B) \#instr = \frac{\text{tiempo CPU} \times \text{tasa reloj}}{CPI} = I \quad \#ciclos = CPI \times \#instr. = C$$

Notar que se está asumiendo mismo CPI de arriba, pero esto podría no ser cierto, dependiendo del programa

$$I_{P_1} = 20 \times 10^9 \quad C_{P_1} = 30 \times 10^9$$

$$I_{P_2} = 25 \times 10^9 \quad C_{P_2} = 25 \times 10^9$$

$$I_{P_3} = 18,2 \times 10^9 \quad C_{P_3} = 40 \times 10^9$$

$$C) \frac{\text{tiempo CPU nuevo}}{\text{tiempo CPU viejo}} = 0,7 = \frac{\frac{\#instr \times CPI_{nuevo}}{\text{tasa reloj nueva}}}{\frac{\#instr. \times CPI_{vieja}}{\text{tasa reloj vieja}}} = \frac{\text{tasa reloj vieja} \times 1,2}{\text{tasa reloj nueva}}$$

$$\text{tasa reloj nueva} = \text{tasa reloj vieja} \times \frac{1,2}{0,7}$$

$$\rightarrow \text{Para } P_1 : 5,14 \text{ GHz}$$

$$\rightarrow \text{Para } P_2 : 4,29 \text{ GHz}$$

$$\rightarrow \text{Para } P_3 : 6,86 \text{ GHz}$$

$$D) \text{ monociclo : } CPI = 1$$

$$\text{Pipeline : 1 burbuja entre lw, add} \rightarrow CPI = \frac{6}{5}$$

$$E) t_{mono} = \frac{5}{\text{tasa reloj mono}} = t_{pipeline} = \frac{6}{\text{tasa reloj pipeline}}$$

$$\rightarrow \text{tasa reloj pipeline} = \frac{6}{5} \times 750 \text{ MHz} = 900 \text{ MHz}$$

P2/ Notar opcode $001000_{bin} = 8_{dec} \rightarrow addi$

- A) Read Reg 1 : $01100_{bin} = 12$
Read Reg 2 : $01101_{bin} = 13$
Write Reg 1 : $01101_{bin} = 13$
Write Data : $\underbrace{10}_{\text{contenido reg. 12}} - \underbrace{20}_{\text{"immediate"}} = -10$

Reg Write : 1

- B) Read Data 1 : 10_{dec}
Read Data 2 : 15_{dec} (no usado)

C) $(PC+4) + \text{"immediate"} \times 4 = 11B8_{hex} + -20_{dec} \times 4 = 4456_{dec}$
 $= 1168_{hex}$

D) $PC+4 = 11B8_{hex}$

E) Reg Dst : 0

ALUSrc : 1

MemtoReg : 0

Jump : 0

Branch AND Zero : 0

F) Se lee registro 8 en vez de 12 \rightarrow Suma ALU : $11 + -20 = -9$

G) Varias opciones. Se puede por ejemplo llenar \$4 con un 1 dec
luego hacer $add \$5, \$4, \$0$. Si hay error
\$5 guardará un 0 en vez de 1.

P3

A) Para S1
 I1 → I2 *is resolve on forwarding?*
 I1 → I3 *Si*
 I2 → I4 *Si*
 I3 → I4 *No*
 I4 → I5 *Si*
 I5 → I6 *Si*
 I5 → I7 *Si*

B)
 Para S1

add IF ID EX M WB

lw → nop IF ID ☐ ☐ ☐

lw → nop ID ☐ ☐ ☐

lw ID EX M WB

lw IF ID EX M WB

and → nop IF ID ☐ ☐ ☐

and → nop ID ☐ ☐ ☐

and ID EX M WB

or → nop IF ID ☐ ☐ ☐

or → nop ID EX M WB

or IF ID ☐ ☐ ☐

sw → nop ID ☐ ☐ ☐

sw → nop IF ID EX M WB

or

Para S2: *is resolve on forwarding?*

lw IF ID EX M WB

and → nop IF ID ☐ ☐ ☐

and → nop ID ☐ ☐ ☐

and ID EX M WB

lw → nop IF ID ☐ ☐ ☐

lw → nop ID ☐ ☐ ☐

lw ID EX M WB

lw IF ID EX M WB

lw

C) Para S1: mismos riesgos > se agrega:
 $I1 \rightarrow I4$
 $I3 \rightarrow I6$
 $I4 \rightarrow I7$ } se resuelven con adelantamiento

Para S2: mismos riesgos

D) Para S1:
 se adelanta $1 + 3 \cdot 1 = 32$, reemplaza 1

Para S2:
 se adelanta 0 (memoria de datos) , reemplaza -2

E)

$$CPI = \frac{N-1 + X \cdot b \cdot p \cdot (i-1) + eX}{eX} \approx \frac{bp(i-1) + e}{e}$$

\uparrow
 $X \gg N$

F) En este caso $b=e$, $p=1$

$$\Rightarrow CPI = i$$

Se requeriría que la comparación de los saltos condicionales se realizara en la segunda etapa del pipeline.