

• Nombre: Tabián

• Apellidos: Scherle Carboneres

11) loop:

- 1- ld (r3), 0 (r2)
- 2- ld (r4), 0 (r3)
- 3- sd (r4), 1024 (r3)
- 4- dadd r1, r1, r4
- 5- daddi (r10), r10, -1
- 6- daddi (r2, r2, 8
- 7- bnez (r10) loop
- 9- <sqte + 1>
- 10- <sqte + 2>
- 11- <sqte + 3>

corrot circuit → datos
predict not taken 2 → control
↳ modifica PC en EX
 $CPI = \frac{10}{7} = 1,43 //$

- 1- IF ID EX M W
- 2- IF ID ID EX M W
- 3- IF IF ID EX M W
- 4- IF ID EX M W

5- IF ID EX M W

6- IF ID EX M W

7- IF ID EX M W

8- IF ID X

9- IF X

↳ aborta las instrucciones

12)

mul.d F2, F0, F0 → IF ID M M M M M M WB

beqz r1, etiqueta → IF ID EX MEM WB

add.d F2, F1, F1 → IF ID ID ID A A A WB
2

13) Predictor 2 bits con saturación:

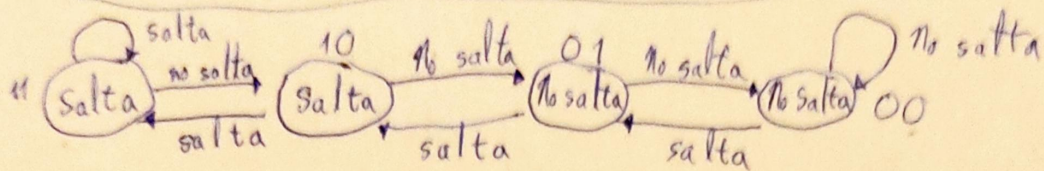


tabla \rightarrow 16 entradas vacías

Fallo de predicción \rightarrow 2 ciclos de parat

predict not taken \rightarrow 3 ciclos de penalización

• bnez t2, sigue

- En la primera iteración la tabla está vacía, se emplea predict not taken, y el salto es efectivo por lo que produce 3 ciclos de penalización, se almacena con Strongly Taken, luego ejecuta 9 iteraciones y en la siguiente $VE_{t2}=1$ por lo que el predictor falla y provoca 2 ciclos de parada, se modifica a Weakly Taken y aún predice que salta, por lo que no producirá mas penalización. $(3 + 2 = 5)$

• bnez t0, loop

- Primera iteración ocurre lo mismo que la primera branch, + 3 ciclos de penalización, y fallará en la última iteración, ya que predice que no será efectivo + 2 ciclos de parada

$$\underline{3 + 2 = 5}$$