

① 3. c) address de p en \$s0

```
LW $T1, 0($s0)
ADDIV $T2, $0, 100
SW $T2, 0($T1)
```

b) address del inicio de a en \$s0

a) address del inicio de a en \$s0

```
ADDIV $T1, $0, 10
ADDIV $T2, $0, 100
Load:
SW $T1, 0($s0)
ADDIV $s0, $s0, 4
ADDIV $T2, $T2, -1
BNE $T2, $0, Load
```

mismo código excepto en una instrucción

----- ADDIV \$s0, \$s0, 1

2. En a y b se utiliza base+offset porque se está guardando en el primer elemento de un array de memoria un valor de otro registro, que en este caso es un immediate correspondiente a I-type.
En la c se está realizando lo mismo de guardar en la address data correspondiente a la edición a, por lo que el base+offset

$$2. \begin{array}{l} \text{A clock rate } 3.6 \text{ Hz} \\ \text{CPI } 1.5 \end{array} \quad \text{IPS}_A = \frac{3.6 \text{ Hz}}{1.5} = 2 \cdot 10^9$$

$$\begin{array}{l} \text{B clock rate } 2.6 \text{ Hz} \\ \text{CPI } 1.0 \end{array} \quad \text{IPS}_B = \frac{2.6 \text{ Hz}}{1.0} = 2.6 \cdot 10^9$$

$$\begin{array}{l} \text{C clock rate } 4.06 \text{ Hz} \\ \text{CPI } 2.2 \end{array} \quad \text{IPS}_C = \frac{4.06 \text{ Hz}}{2.2} = 1.845 \cdot 10^9$$

1* A y B tienen la performance más alta, siendo la misma en ambas

tiempo 10 segundos

$$2* \begin{array}{l} \text{A} \\ \text{instrucciones} = \text{IPS}_A \cdot \text{CPUT} = 2 \cdot 10^9 \cdot 10 = 2 \cdot 10^{10} \\ \text{clock cycles} = \text{CPUT} \cdot \text{clock rate}_A = 10 \cdot 3.6 \cdot 10^9 = 3.6 \cdot 10^{10} \end{array}$$

$$\begin{array}{l} \text{B} \\ \text{instrucciones} = \text{IPS}_B \cdot \text{CPUT} = 2.6 \cdot 10^9 \cdot 10 = 2.6 \cdot 10^{10} \\ \text{clock cycles} = \text{CPUT} \cdot \text{clock rate}_B = 10 \cdot 2.6 \cdot 10^9 = 2.6 \cdot 10^{10} \end{array}$$

$$\begin{array}{l} \text{C} \\ \text{instrucciones} = \text{IPS}_C \cdot \text{CPUT} = 1.845 \cdot 10^9 \cdot 10 = 1.845 \cdot 10^{10} \\ \text{clock cycle} = \text{CPUT} \cdot \text{clock rate}_C = 10 \cdot 4.06 \cdot 10^9 = 4.06 \cdot 10^{10} \end{array}$$

$$\text{tiempo de ejecución} = 0.7 \text{ tiempo de ejecución prev} \quad \text{CPI} = 1.2 \text{ CPI prev}$$

$$\frac{\text{instrucciones} \cdot \text{CPI}}{\text{clock rate}} = 0.7 \frac{\text{instrucciones prev} \cdot \text{CPI prev}}{\text{clock rate prev}}$$

$$\frac{\text{CPI}}{\text{clock rate}} = 0.7 \frac{\text{CPI prev}}{\text{clock rate prev}}$$

$$\frac{1.2 \text{ CPI prev}}{\text{clock rate}} = 0.7 \frac{\text{CPI prev}}{\text{clock rate prev}}$$

$$\frac{1.2}{\text{clock rate}} = \frac{0.7}{\text{clock rate prev}}$$

$$3* \text{clock rate} = 1.714 \cdot \text{clock rate prev}$$

$$3. \left\{ 1 \cdot 10^5 + 2 \cdot 2 \cdot 10^5 + 3 \cdot 5 \cdot 10^5 + 3 \cdot 2 \cdot 10^5 = 2,6 \cdot 10^6 \right.$$

$$P1 \left\{ \frac{2,6 \cdot 10^6}{2,5 \text{ GHz}} = 1,04 \text{ ms} \quad \text{CPVT} \right.$$

$$P2 \left\{ 2 \cdot 10^5 + 2 \cdot 2 \cdot 10^5 + 2 \cdot 5 \cdot 10^5 + 2 \cdot 2 \cdot 10^5 = 2 \cdot 10^6 \right.$$

$$\frac{2 \cdot 10^6}{3 \text{ GHz}} = 666,67 \text{ ns} \quad \text{CPVT}$$

① * P2 tiene una implementación más rápida

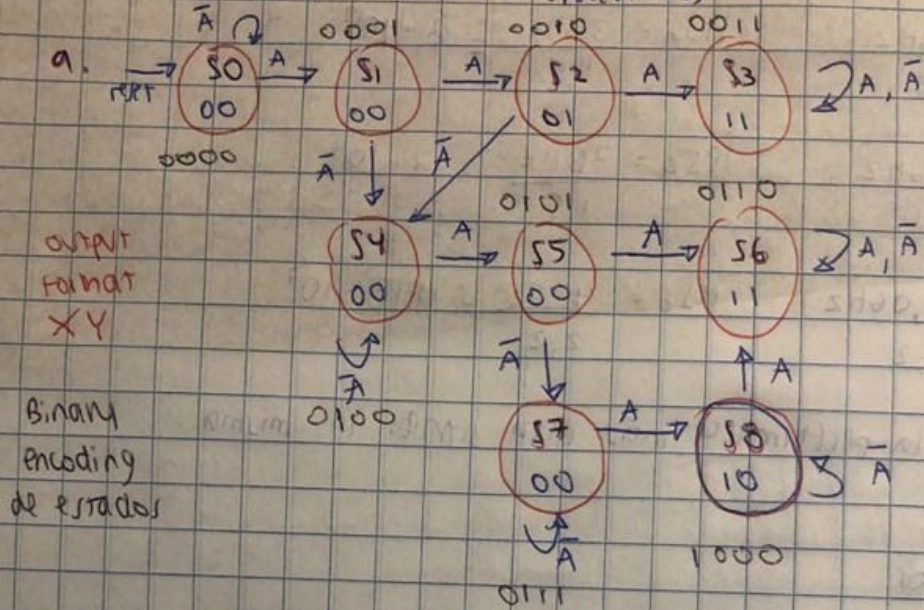
$$② * \frac{2,6 \cdot 10^6}{10^6} = 2,6 \quad P1 \quad \text{CPI}$$

$$* \frac{2 \cdot 10^6}{10^6} = 2 \quad P2 \quad \text{CPI}$$

$$③ * 2,6 \cdot 10^6 \quad \text{cycles} \quad P1$$

$$* 2 \cdot 10^6 \quad \text{cycles} \quad P2$$

4. input A
 output X
 output Y
 A=1 tres veces (no necesariamente consecutivas)
 A=1 dos veces (consecutivas)



b.

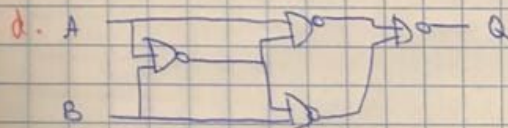
S	A	S'
0000	0	0000
0000	1	0001
0001	0	0100
0001	1	0010
0010	0	0100
0010	1	0011
0011	X	0011
0100	0	0100
0100	1	0101
0101	0	0111
0101	1	0110
0110	X	0110
0111	0	0111
0111	1	1000
1000	0	1000
1000	1	0110

c.

5. c. Un latch cambia si es que los valores han seguido cambiando y los
 toca cambiar las asignaciones w_0 , mediante de Q y $\sim Q$, por ende
 también los inputs del RSLatch. Es así mismo a un Enable en el caso de estar
 un FlipFlop en cambio cambia cada vez que hay un clock edge.
 Es síncrono al clk.

→ se puede usar un FF si se quieren representar estados, los valores
 pueden cambiar únicamente luego de un ciclo del reloj

→ se pueden usar LATCHES si es que lo importante es utilizarlos como
 registros. por eso cuando toques la señal de escribir se guardará
 la data



El mínimo que se puede usar
 es 4

b. 1. Microarchitecture
 2. ISA

a. ~~Diagrama de conexión de la red~~
 un jump que dirige al PC a la dirección
 previa (o sea a sí misma) para que
 no incremente en 1.