

ARQUITECTURA E INGENIERÍA DE COMPUTADORESEjercicios propuestos del Tema 2: Segmentación y superescalares**1. Dada la siguiente tabla de reservas**

	1	2	3	4	5
S <sub>1</sub>	x	x			x
S <sub>2</sub>	y		x	y	y
S <sub>3</sub>	x			x	y

y disponiendo de la secuencia yxxyyx:

a) Calcular el número de ciclos que tarda en ejecutarse esta secuencia (T<sub>SEC</sub>).

En primer lugar, calculamos las latencias prohibidas:

$$F_{xx} = \{1, 3, 4\}$$

$$F_{xy} = \{1, 2, 4\}$$

$$F_{yx} = \{2\}$$

$$F_{yy} = \{1, 3, 4\}$$

Posteriormente, calculamos los vectores de colisiones cruzadas y las matrices de colisión:

$$V_{xx} = (1101)$$

$$V_{xy} = (1011)$$

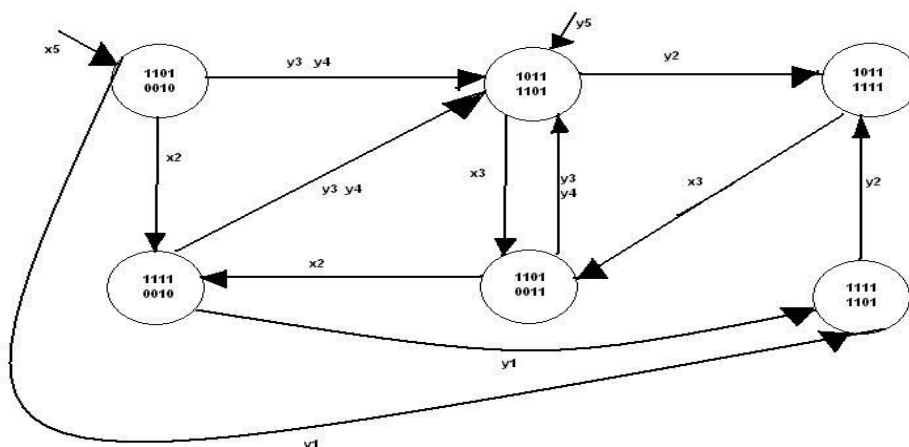
$$V_{yx} = (0010)$$

$$V_{yy} = (1101)$$

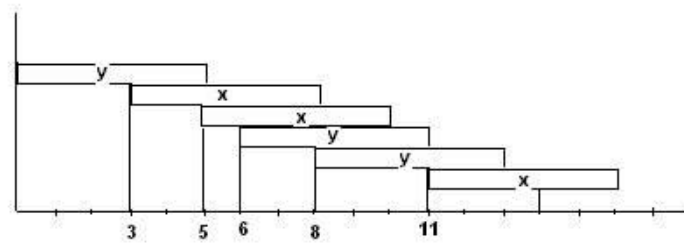
$$M_x = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$M_y = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Una vez tenemos estos datos, podemos construir el diagrama de estados:



Con el diagrama de estados construido, podemos construir la siguiente tabla:



Con esto podemos sacar el número de ciclos que tarda la ejecución de la secuencia:

$$T_{\text{SEC}} = 3 + 2 + 1 + 2 + 3 + 5 = 16 \text{ ciclos}$$

**b) Calcula el rendimiento máximo del cauce ( $W_{\text{max}}$ ), suponiendo que la CPU tiene una frecuencia de 1 GHz.**

Podemos definir el rendimiento máximo como

$$W_{\text{max}} = \lim_{n \rightarrow \infty} \frac{n}{T(n)} = \frac{1}{14} \text{ ciclos}^{-1}$$

siendo  $T(n)$  14 porque es el tiempo que tarda en poder entrar la siguiente secuencia.

Despejando la fórmula del rendimiento máximo, obtenemos:

$$W_{\text{max}} = \frac{1 \text{ secuencia} * 6 \text{ instrucciones/secuencia}}{14 \text{ ciclos} * 10^{-9} \text{ segundos/ciclo}} = 428,5714 \text{ MOPS}$$

2. Dado un programa que tiene una instrucción de salto con el siguiente comportamiento (siendo la S que sí se salta y la N que no se salta):

**S N S S S N N S N N S S N N N S S**

Calcular la penalización efectiva con las predicciones de salto, si la penalización por un fallo en la predicción es de 4 ciclos:

**a) Estática TAKEN**

En este caso, siempre que no se salte se producirá una penalización, entonces tenemos:

S	N	S	S	S	N	N	S	N	N	S	S	N	N	N	S	S
	P				P	P		P	P			P	P	P		

Tenemos penalización en 8 ocasiones, luego la penalización es de:

$$8P * 4 \text{ ciclos} = 32 \text{ ciclos de penalización}$$

**b) Estática NOT-TAKEN**

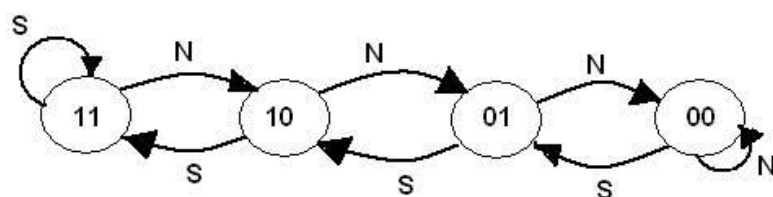
En este caso, la penalización se llevará a cabo siempre que se salte, teniendo lo siguiente:

S	N	S	S	S	N	N	S	N	N	S	S	N	N	N	S	S
P		P	P	P			P			P	P				P	P

Tenemos penalización en 9 ocasiones, luego la penalización es de:

$$9P * 4 \text{ ciclos} = 36 \text{ ciclos de penalización}$$

**c) Predicción dinámica con 2 bits de historia (estado inicial 11)**



Tenemos la siguiente secuencia:

<b>Acción</b>	S	N	S	S	S	N	N	S	N	N	S	S	N	N	N	S	S
<b>Penal.</b>		P				P	P	P	P		P	P	P			P	P
<b>Pred.</b>	S	S	S	S	S	S	S	N	S	N	N	N	S	N	N	N	N

Obtenemos penalización en 10 ocasiones, luego los ciclos de penalización son:

$$10P * 4 \text{ ciclos} = 40 \text{ ciclos de penalización}$$

**d) Predicción dinámica con 3 bits de historia, inicialmente todos a 1**

Con 3 bits de historia obtenemos el siguiente resultado:

Acción	Penalización	Predicción	Bits
S		S	111
N	P	S	111
S		S	011
S		S	101
S		S	110
N	P	S	111
N	P	S	011
S	P	N	001
N		N	100
N		N	000
S	P	N	000
S	P	N	100
N	P	S	110
N	P	S	011
N		N	001
S	P	N	000
S	P	N	100

Obtenemos también 10 penalizaciones, con el siguiente resultado:

$$10P * 4 \text{ ciclos} = 40 \text{ ciclos de penalización}$$