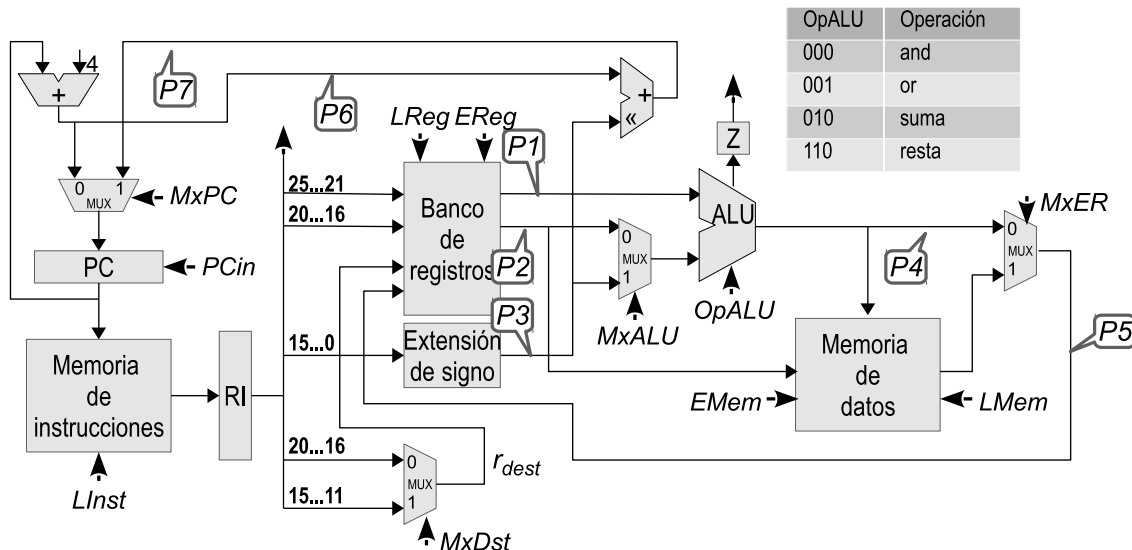


Apellidos y nombre

DNI

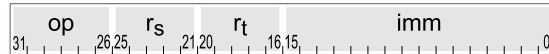
Grupo

1 (1.5 puntos) Considere la ruta de datos monociclo:



- a) (0.8 puntos) La siguiente tabla muestra diversos estados de la ruta de datos definidos por los campos del registro RI y las señales de control que se aplican. Deduzca, en cada caso, a qué instrucciones del juego del MIPS se refieren. Todas son del formato I.

Formato I:



RI			Señales								Instrucción
25...21	20...16	15..0	MxALU	OpALU	LMem	EMem	Ereg	MxDst	MxER	MxPC	
2	4	8	1	000	1	0	0	0	0	0	andi \$4,\$2,8
2	4	8	1	010	0	0	1	0	0	0	addi \$4,\$2,8
2	4	8	1	010	0	1	0	0	0	0	sw \$4,8(\$2)
2	4	8	1	010	1	0	1	0	1	0	lw \$4,8(\$2)
2	4	8	1	001	0	0	1	0	0	0	ori \$4,\$2,8

- b) (0,7 puntos) Suponiendo que el contenido de algunos registros y de la memoria del MIPS es el mostrado a la derecha, indique (en decimal) el valor presente en los puntos rotulados en la ruta de datos como P1 a P7. Considere en todos los casos que la instrucción se encuentra en la dirección 1000 (decimal), o sea, que el PC contiene la dirección 1000. Si en algún caso falta información, indíquelo con "???"

Registros

Reg	Contenido
\$2	200
\$3	300
\$4	400
\$5	500

Instrucción	P1	P2	P3	P4	P5	P6	P7
sw \$3,20(\$2)	200	300	20	220	???	1004	1084
lw \$3,20(\$2)	200	300	20	220	17	1004	1084
beq \$2,\$3,2004	200	300	250	-100	???	1004	2004

Memoria principal

Dir	Contenido
220	17

**2** (1 punto) En la ruta de datos anterior, suponga que operar con la memoria de instrucciones requiere 5 ns, operar con la memoria de datos 6 ns, leer o escribir en el banco de registros 1 ns y operar en la ALU 7 ns. El tiempo necesario para decodificar las instrucciones y generar las señales de control es despreciable. Además, el procesador tiene que ejecutar un programa  $P$  compuesto por 100 millones de instrucciones. Indique, **justificando SIEMPRE la respuesta**:

a) (0.5 puntos) ¿Cuál es la máxima frecuencia de reloj a la que puede trabajar este procesador monociclo?

$$\text{Període} = 5 + 1 + 7 + 6 + 1 = 20 \text{ ns}$$

$$f = 50 \text{ MHz}$$

El tiempo de ejecución del programa  $P$  en segundos

$$T = 100 \times 10^6 \times 20 \times 10^{-9} = 2 \text{ seg}$$

b) (0.5 puntos) Para aumentar su productividad, se segmenta la ruta de datos en las **CUATRO** etapas siguientes:

- LDI: Leer la instrucción, decodificarla y leer los registros.
- EX: Operar en la UAL
- M: Acceder a la memoria de datos
- ER: Escribir el resultado en el banco de registros

¿Cuál será el retardo de la etapa LDI?

$$5 + 1 = 6 \text{ ns}$$

¿Cuál será la etapa más lenta?

$$\text{ALU, } 7 \text{ ns}$$

Si el retardo de los registros de segmentación es de 1 ns, ¿cuál será la máxima frecuencia a la que puede funcionar el reloj del procesador segmentado?

$$\text{període mínim} = 8 \text{ ns}; f = 125 \text{ MHz}$$

c) (0,5 puntos) Como este procesador segmentado no detecta los conflictos de datos, los compiladores han de añadir instrucciones NOP al código. El procesador resuelve los conflictos de control mediante ciclos de parada. Por esa razón, al programa  $P_S$  (versión de  $P$  para este procesador) se le añaden 20 millones de instrucciones NOP y su ejecución produce 30 millones de ciclos de parada.

¿Cuál será el valor del CPI?

$$\text{CPI} = (120 + 30) \text{ M. de ciclos} / 120 \text{ M. d'instruccions} = 1.25$$

¿Cuál será el tiempo de ejecución del programa  $P_S$  en segundos?

$$T = 120 \times 10^6 \times 1.25 \times 8 \times 10^{-9} = 1.2 \text{ seg}$$

**3** (1 punto) En el procesador segmentado en las cinco etapas habituales (LI, DI, EX, M, ER) se va a ejecutar el siguiente fragmento de código en ensamblador del MIPS R2000.

```
(1)      lw $t2,0($t1)
(2)      addi $t1,$t1,4
(3)      add $t2,$t2,$t4
(4)      sub $t4,$t4,$t5
(5)      sw $t2,0($t1)
```

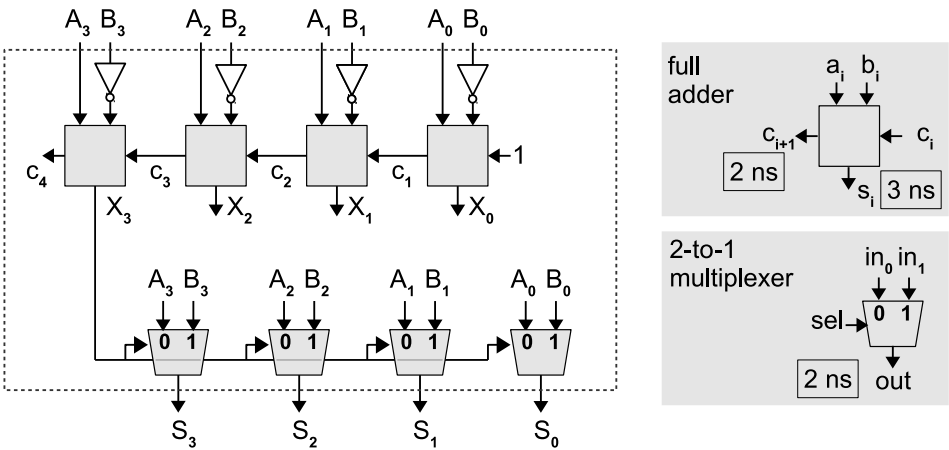
a. (0,5 puntos) Señale en la tabla siguiente TODAS las dependencias de datos presentes en el código. En la columna de la derecha, indique para cada dependencia si aparece conflicto de segmentación o no

	Registro	Instrucción en que se escribe	Instrucción en que se lee	Conflicto? Si/No
Riesgo 1	\$t2	1	3	Sí
Riesgo 2	\$t2	3	5	Sí
Riesgo 3	\$t1	2	5	No
Riesgo 4				
Riesgo 5				

b. (0.5 puntos) Complete el diagrama de instrucciones/ciclo suponiendo que los conflictos por dependencias de datos se resuelven mediante la inserción de ciclos de parada.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
lw \$t2,0(\$t1)	LI	DI	EX	M	ER													
addi \$t1,\$t1,4		LI	DI	EX	M	ER												
add \$t2,\$t2,\$t4			LI	DI	DI	EX	M	ER										
sub \$t4,\$t4,\$t5				LI	LI	DI	EX	M	ER									
sw \$t2, 0(\$t1)						LI	DI	DI	EX	M	ER							

**4** (2 puntos) La figura muestra un operador que acepta dos operandos A y B de 4 bits, de tipo entero con signo, y devuelve el mayor de ellos  $S = \max\{A,B\}$ . Note que la parte superior (formada por cuatro sumadores completos y cuatro puertas NOT) realiza una resta  $X=A-B$  y que la parte inferior (formada por los multiplexores) selecciona entre A y B en función del signo de X. El operador sólo funciona correctamente si no hay desbordamiento en la resta  $A-B$ .



- a. (0,4 puntos) Complete la tabla siguiente con los valores de X y S (cuatro bits cada uno) y los bits de transporte  $c_4$  y  $c_3$ :

Caso	A	B	X	$c_4$	$c_3$	S
1	0101	0011	0010	1	1	0101
2	0011	0101	1110	0	0	0101
3	1111	1110	0001	1	1	1111
4	0011	1100	0111	0	0	0011
5	1010	0110	0100	1	0	1010

- b. (0,3 puntos) A la vista de los datos de la tabla, ¿ha habido algún desbordamiento? ¿Cómo se detecta?

*En el caso 5 hay desbordamiento y se nota porque  $c_4 \neq c_3$ . El signo de X es incorrecto y por eso el multiplexor selecciona el operando equivocado*

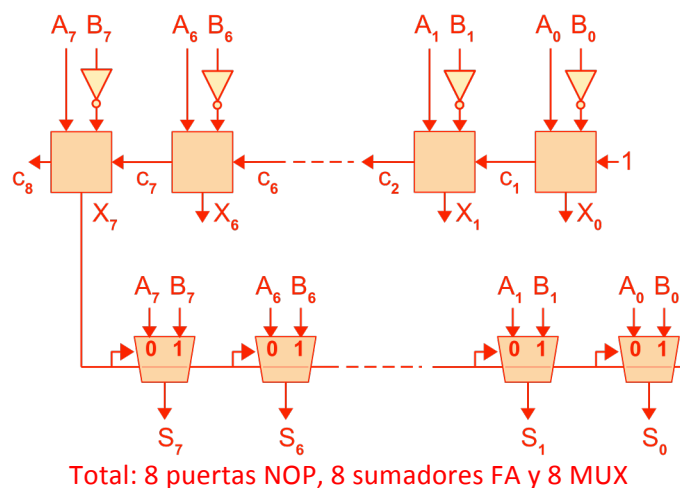
- c. (0,8 puntos) Teniendo en cuenta los retardos de los componentes mostrados en la figura y que el retardo de una puerta NOT es de 1 ns, calcula el instante en que son válidos los bits que se indican a partir del momento en que se aplican las entradas

valor	$c_1$	$c_3$	$X_3$	$S_3$	$S_0$
retardo (ns)	3	7	10	12	12

¿Cuál es la productividad del operador en MOPS?

*Retardo=12 ns; Productividad=83,3 MOPS*

- d. (0,5 puntos) ¿Cuántos componentes de cada tipo (sumadores Full Adder, multiplexores y puertas NOT) harían falta para un operador semejante que operara con números de 8 bits? Haga un esquema lo más simple que sea posible del circuito.



5 (0,5 puntos) Escriba en ensamblador el código de una función que en alto nivel se expresa así:

```
float prodWF(int a, float x){
    return a*x;
}
```

Note que la función realiza el producto entre un valor entero y un valor de coma flotante. En ensamblador, la función recibe el valor entero (*a*) en el registro \$a0 y el valor de coma flotante (*x*) en el registro \$f10 y ha de devolver el resultado en \$f0.

```
prodWF: mtc1 $a0,$f0
        cvt.s.w $f0,$f0
        mult.s $f0,$f0,$f10
        jr $ra
```

6 (1.5 puntos) Considere un chip de memoria SDRAM de 64-bit de ancho de palabra con los parámetros temporales mostrados en la tabla siguiente. Suponga que el chip está conectado a un bus que trabaja a una frecuencia de reloj de 200 MHz (*t*<sub>ciclo</sub> = 5 ns).

a. (0.3 puntos) Complete la tabla anotando el número de ciclos correspondientes a cada parámetro.

Parámetro	ns	Ciclos
<i>t</i> <sub>rcd</sub>	12	$\lceil 12/5 \rceil = 3$
<i>t</i> <sub>ras</sub>	35	$\lceil 35/5 \rceil = 7$
<i>t</i> <sub>rc</sub>	50	$\lceil 50/5 \rceil = 10$
<i>t</i> <sub>rp</sub>	10	$\lceil 10/5 \rceil = 2$
CL	—	3

b. (0.3 puntos) Dibuje el cronograma de una operación de lectura de dos bloques de 4 palabras localizados en la misma fila de la matriz de memoria. Suponga precarga automática.

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27
Comando	ACT			RD				RD																			
Dirección	fila			col1 w				col2 w'																			
Datos								w <sub>0</sub>	w <sub>1</sub>	w <sub>2</sub>	w <sub>3</sub>	w <sub>0</sub> '	w <sub>1</sub> '	w <sub>2</sub> '	w <sub>3</sub> '												

c. (0.3 puntos) Calcule el ancho de banda del chip.

64 bits \* 200 MHz = 8 B \* 200 MHz = 1600 MBps = 1.6 GBps

d. (0.3 puntos) Calcule el tiempo de acceso de lectura.

$t_{RCD} + CL = 3 + 3 \text{ ciclos} = 6 \text{ ciclos}; 6 \text{ ciclos} * 5 \text{ ns} = 30 \text{ ns}$

e. (0.3 puntos) Considere ahora un chip de memoria con parámetros temporales idénticos pero con interfaz DDR2. Calcule el ancho de banda y el tiempo de acceso de lectura del chip.

El mismo tiempo de acceso ( $t_{RCD} + CL$ ). El doble ancho de banda de  $2 * 1.6 \text{ GBps} = 3.2 \text{ GBps}$

**7** (1.5 puntos) Se está diseñando un mapa de memoria para un computador basado en un MIPS R2000.

- Un módulo M1 de 256 MB de capacidad, ubicado en las direcciones superiores del mapa.

a. (0.25 puntos) ¿Cuál sería (en hexadecimal) la dirección inferior contenida en este módulo?

$256 \text{ M} = 2^{28} \rightarrow 32 - 28 = 4$ . Los 4 bits más significativos de la dirección valdrán 1 para alcanzar las direcciones más altas. Por lo tanto, la dirección más baja será  $0xF0000000$

b. (0.25 puntos) Determine la función de selección para este módulo, suponiendo lógica activa por nivel alto.

$f(M1) = \neg(A_{31} \cdot A_{30} \cdot A_{29} \cdot A_{28})$

- Otro módulo, M2, ocupa el intervalo de direcciones desde  $0x00000000$  hasta  $0x3FFFFFFF$ .

c. (0.25 puntos) ¿Cuál es la capacidad de este módulo de memoria?

Si varían 30 bits de la dirección  $\rightarrow 2^{30} = 1 \text{ GB}$

d. (0.25 puntos) ¿Cuántas palabras se pueden almacenar en M2?

Palabras de 32 bits  $\rightarrow 4 \text{ bytes/palabra}; 1 \text{ GB} / 4 \text{ B} = 256 \text{ M palabras}$

- Se ubica un módulo M3 de 512 MB a partir de la dirección  $0xC0000000$

e. (0.25 puntos) ¿Cuál es la dirección (en hexadecimal) más alta localizada en M3?

$512 \text{ M} = 2^{29} \rightarrow 0xC0000000 + 0x1FFFFFFF = 0xDFFFFFFF$

f. (0.25 puntos) Indique los rangos de direcciones no cubierto por los módulos M1, M2 y M3 en el mapa de memoria.

Rangos de direcciones:  $0x40000000..0xBFFFFFFF$  (2 GB) y  $0xE0000000..0xFFFFFFFF$  (256 MB)