

Nombre y Apellidos

DNI

Grupo

--	--	--

- 1 (1.5 puntos)** Se ha modificado la ruta de datos estudiada en clase para dar soporte a las tres instrucciones de desplazamiento clásicas: a la izquierda («), lógico a la derecha (»_L) y aritmético a la derecha (»_A). En este nuevo diseño el valor a desplazar siempre está en *Rs*. Para cada operación hay dos instrucciones: una de desplazamiento constante, donde el número de bits a desplazar (*sa*, de *shift amount*, $0 \leq sa \leq 31$) está codificado en los 5 bits inferiores del campo inmediato (instrucción tipo I), y la de desplazamiento variable, que obtiene el número de bits a desplazar de los cinco bits inferiores del registro *Rt* (instrucción tipo R). En total, son seis instrucciones.

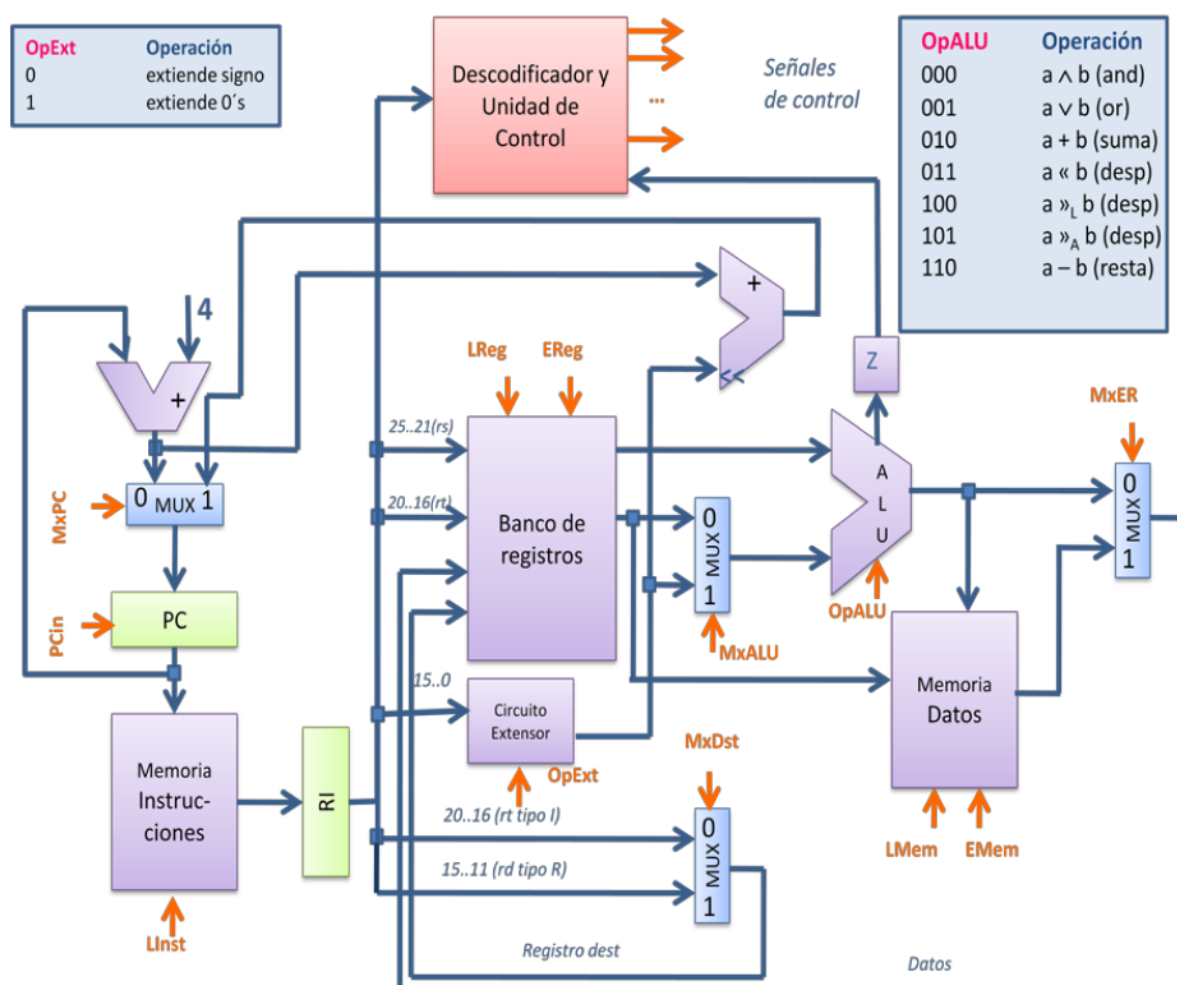
instrucción	operación
sll <i>rt,rs,sa</i>	$rt = rs \ll sa$
srl <i>rt,rs,sa</i>	$rt = rs \gg_L sa$
sra <i>rt,rs,sa</i>	$rt = rs \gg_A sa$

instrucción	operación
sllv <i>rd,rt,rs</i>	$rd = rs \ll rt$
srlv <i>rd,rt,rs</i>	$rd = rs \gg_L rt$
srav <i>rd,rt,rs</i>	$rd = rs \gg_A rt$

Para ello, a la ALU se la ha dotado de operaciones adicionales:

OpALU	Operación
011	«, desplazamiento izquierda
100	» _L , desplazamiento lógico derecha
101	» _A , desplazamiento aritmético derecha

Además, a la ruta de datos se le ha añadido un circuito para la extensión de 16 a 32 bits del operando inmediato de las operaciones aritméticas y lógicas de tipo I. El circuito se controla mediante la señal *OpExt* tal como indica la tabla.



Complete la tabla de señales de control para cada instrucción:

Instrucción	EReg	OpExt	MxDst	MxALU	OpALU	LMem	EMem	MxER	MxPC
sll	1	x	0	1	011	0	0	0	0
srav	1	x	1	0	101	0	0	0	0
bne	0	0	x	0	110	0	0	x	Z*
sw	0	0	x	1	010	0	1	x	0
add	1	x	1	0	010	0	0	0	0
addi	1	0	0	1	010	0	0	0	0
ori	1	1	0	1	001	0	0	0	0

2 (1.5 puntos) En el procesador monociclo de la pregunta anterior las operaciones en memoria conllevan 15 ns, leer y escribir en el banco de registros 5 ns, el circuito multifunción 3ns y operar en la ALU 10 ns. Suponga que el resto de elementos tienen un retardo despreciable.

Para aumentar la productividad de este procesador se procede a segmentarlo en las 5 etapas vistas en clase (LI, DI, EX, M, ER). El retardo de los registros de segmentación es de 3ns.

Para estudiar las prestaciones de ambos procesadores se ejecutan dos programas A y B, con un millón de instrucciones cada uno. En el segmentado, el primer programa tiene un CPI=2 y el segundo programa un CPI=1.5.

Rellene los datos solicitados a continuación, **incluyendo el cálculo correspondiente y las unidades.**

TABLA 1 (0,8 puntos)

	Procesador Monociclo	Procesador Segmentado
Tiempo de ciclo	$15 + \max(5, 3) + 10 + 15 + 5 = 50\text{ns}$	$\text{Max}(15, \max(5, 3), 10, 15, 5) + 3 = 18\text{ns}$
Frecuencia	$1/50\text{ns} = 20\text{MHz}$	$1/18\text{ns} = 55,55\text{ MHz}$
Tiempo de ejecución del programa A	$50\text{ns} * 10^6 = 50\text{ms}$	$18\text{ns} * (10^6 * 2 + 4) \approx 36\text{ms}$
Tiempo de ejecución del programa B	$50\text{ns} * 10^6 = 50\text{ms}$	$18\text{ns} * (10^6 * 1.5 + 4) \approx 27\text{ms}$

TABLA 2 (0,40 puntos)

	ACELERACION del segmentado respecto al monociclo
Ideal	5
Máxima	50ns/18ns=2,78
Al ejecutar el programa A	50ms/36ms=1,39
Al ejecutar el programa B	50ms/27ms=1,85

Se plantea mejorar el procesador segmentado, super-segmentándolo en 7 etapas. Contesta a las siguientes cuestiones **justificando las respuestas**:

TABLA 3 (0,30 puntos)

¿Cuál sería la ACELERACION ideal respecto al moniciclo?	7 es el número de etapas
¿Qué etapas super-segmentarías?	LI y M (las dos de memoria por ser las más lentas) lo que permitirá disminuir el ciclo de reloj
¿Cuál sería la productividad máxima?	1 instrucción/ciclo = $1/(10+3)ns = 76.92$ MIPS Considerando $I \rightarrow \infty$, y ausencia de conflictos

3 (1 punto) En el procesador segmentado con 5 etapas del ejercicio anterior se va a ejecutar el siguiente fragmento de código en ensamblador del MIPS R2000. Asuma que se utiliza inserción de ciclos de parada tanto para solucionar los conflictos por dependencias de datos como los riesgos de control. La latencia de salto para este procesador es de 1 ciclo.

```

(1)          addi $t2, $zero, 20
(2) bucle:   lw $t3, 0($t0)
(3)          lw $t4, 0($t1)
(4)          add $t3, $t3, $t4
(5)          sw $t3, 0($t0)
(6)          addi $t0, $t0, 4
(7)          addi $t2, $t2, -1
(8)          addi $t1, $t1, 4
(9)          bne $t2, $zero, bucle
(10)         addi $t2, $zero, 30

```

a) (0.25 puntos) Identifique los conflictos por dependencia de datos que se producen en dicho código, así como el número de ciclos de parada que se insertarán durante su ejecución para resolverlos. Rellene para ello la tabla siguiente utilizando tantas filas como necesite:

	Registro	Número de instrucción en que se escribe	Número de instrucción en que se lee	Ciclos de parada a insertar
Riesgo 1	\$t3	2	4	0
Riesgo 2	\$t4	3	4	2
Riesgo 3	\$t3	4	5	2
Riesgo 4	\$t2	7	9	1

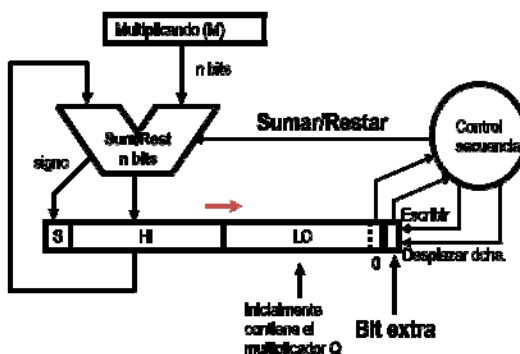
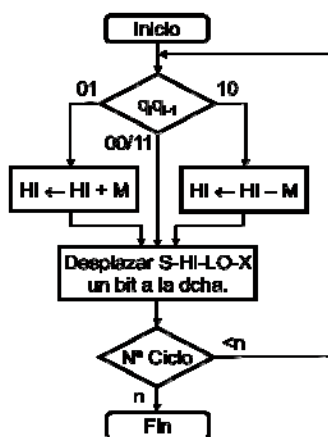
b) (0.5 puntos) Complete el diagrama instrucciones/tiempo para este fragmento de código reflejando lo que ocurre en las **10 primeras instrucciones ejecutadas**.

Instrucción/ciclo	x	x+1	x+2	x+3	x+4	x+5	x+6	x+7	x+8	x+9	x+10	x+11	x+12	x+13	x+14	x+15	x+16	x+17	x+18	x+19
addi	LI	DI	EX	M	ER															
lw		LI	DI	EX	M	ER														
lw			LI	DI	EX	M	ER													
add				LI	--	--	DI	EX	M	ER										
sw							LI	--	--	DI	EX	M	ER							
addi										LI	DI	EX	M	ER						
addi											LI	DI	EX	M	ER					
addi												LI	DI	EX	M	ER				
bne													LI	--	DI	EX	M	ER		
lw															--	LI	DI	EX	M	ER

c) (0.25 puntos) Indique, para la ejecución completa del código del enunciado, y **justificando siempre los resultados**:

Número total de instrucciones ejecutadas (I)	$1 + 20 \cdot 8 + 1 = 162$
Número de ciclos totales de parada (P)	$(5 \text{ DATOS} + 1 \text{ CONTROL}) \cdot 20 = 120$
Número total de ciclos de ejecución (T)	$162 + 120 + 4 = 286$
CPI (indique las operaciones realizadas para el cálculo)	$(286 - 4) / 162 = 1,74$

- 4 (1 punto) Se dispone de un multiplicador secuencial para números con signo de 8 bits cuyo control, mostrado a continuación, implementa el algoritmo de Booth:



Se desea realizar la multiplicación mediante el correspondiente operador hardware de los números $M=2$ y $Q=-9$, donde M representa el multiplicando y Q el multiplicador.

- a) (0.15 puntos) Indique la inicialización de los registros HI y LO en el circuito de multiplicación.

$HI = 00000000$

$LO = Q = -9 = \text{c.a.2 de } +9 = \text{c.a.2 } (00001001) = 11110110 + 1 = 11110111$

- b) (0.15 puntos) Indique la codificación Booth de Q .

$000-1100-1$

- c) (0.2 puntos) Indique las acciones a realizar en el 5º ciclo del algoritmo (sin tener en cuenta el ciclo de inicialización).

$HI \leftarrow HI - M$

Desplazar S-HI-LO 1 bit a la derecha

- d) (0.5 puntos) Calcule la productividad del operador teniendo en cuenta los retardos de las operaciones. Se asume que en un ciclo de reloj se completa una iteración. La inicialización se realiza en un ciclo de reloj adicional.

Desplazar M 1 bit	Inspeccionar q_i, q_{i-1}	suma	resta	escritura HI	Desplazamiento S-HI-LO-X 1 bit	Evaluar Nº ciclo
2 ns	3 ns	15ns	20ns	2 ns	2 ns	5 ns

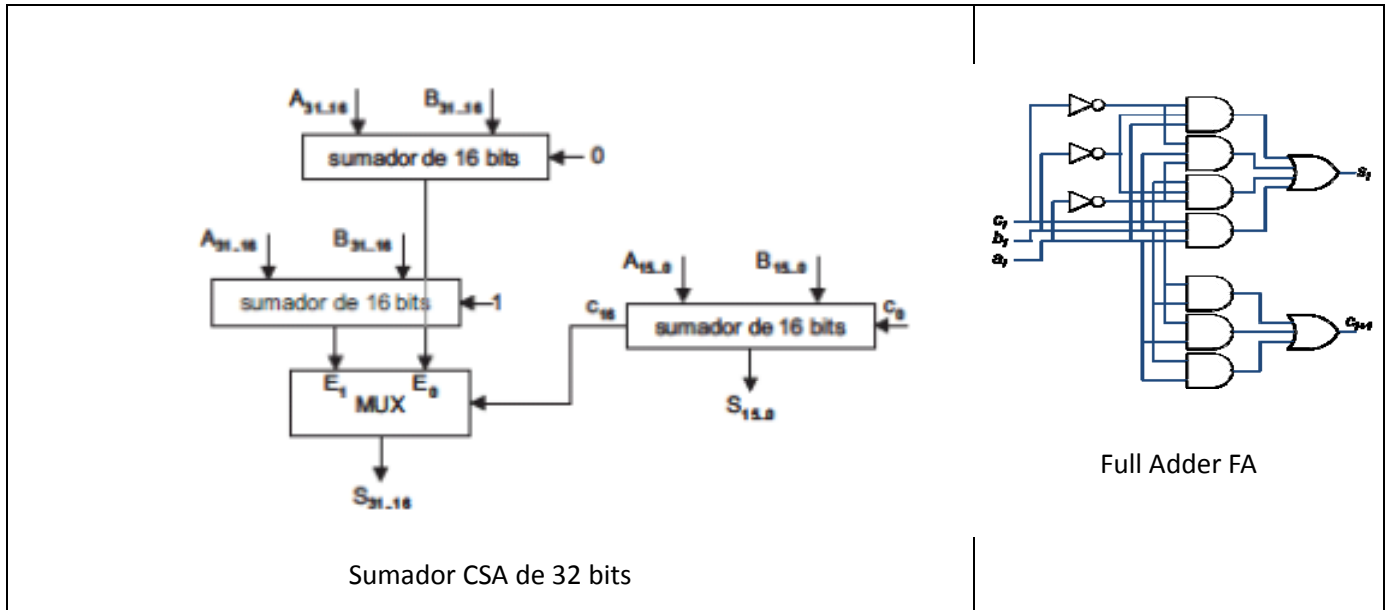
Número de ciclos = $8 + 1$ (inicialización) = 9

Duración mínima de un ciclo = $t_c = t_{\text{insp}} + t_{\text{resta}} + t_{\text{escritura}} + t_{\text{desplazar 1 bit}} + t_{\text{evaluar}} = 3 + 20 + 2 + 2 + 5 = 32\text{ns}$

Duración multiplicación = $T = 9 \times t_c = 9 \times 32 = 288\text{ns}$

Productividad = $1\text{op}/288\text{ns} = 3,47 \text{ MOPS}$

- 5 (1 punto)** Se desea realizar la suma de dos números enteros de 32 bits en complemento a dos mediante la utilización de un sumador **CSA (Carry Select Adder)** como el de la figura. Suponga que los sumadores de 16 bits son CPA (sumadores con acarreo serie), y que están constituidos por FA (Full Adder) diseñados a partir de las funciones lógicas de suma y carry. Considere el retardo de una puerta lógica = 1 ns, y el retardo del multiplexor = 3 ns.



- a) **(0.5 puntos)** Indique el retardo en las siguientes señales y la productividad del sumador CSA:

$T(S_{15}) = \text{retardo de la suma de un CPA de 16 bits} = 2 \times 15 + 3 = 33 \text{ ns}$

$T(C_{16}) = \text{retardo del carry de un CPA de 16 bits} = 2 \times 16 = 32 \text{ ns}$

$T(S_{31}) = 33 + 3 = 36 \text{ ns}$

Productividad = $1/36 \text{ ns} = 1000/36 \text{ MOPS} = 27.78 \text{ MOPS}$

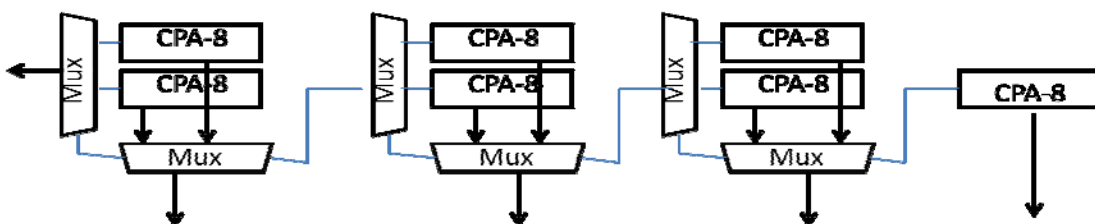
- b) **(0.5 puntos)** Compare el tiempo y la productividad con un sumador CSA diseñado con 4 bloques de 8 bits (8-8-8-8). Los bloques de 8 bits se diseñan a partir de CPAs de 8 bits y los multiplexores necesarios.

Retardo (CSA 8-8-8-8) = **22 ns**

Los CPA de 8 bits tienen un retardo de $2 \times 7 + 3 = 17 \text{ ns}$, y actúan todos en paralelo. Al bloque de más peso le llega la señal con un retardo = 16 (acarreo del CPA de menos peso) + 3×3 (cadena de 3 Mux en serie) = 25 ns

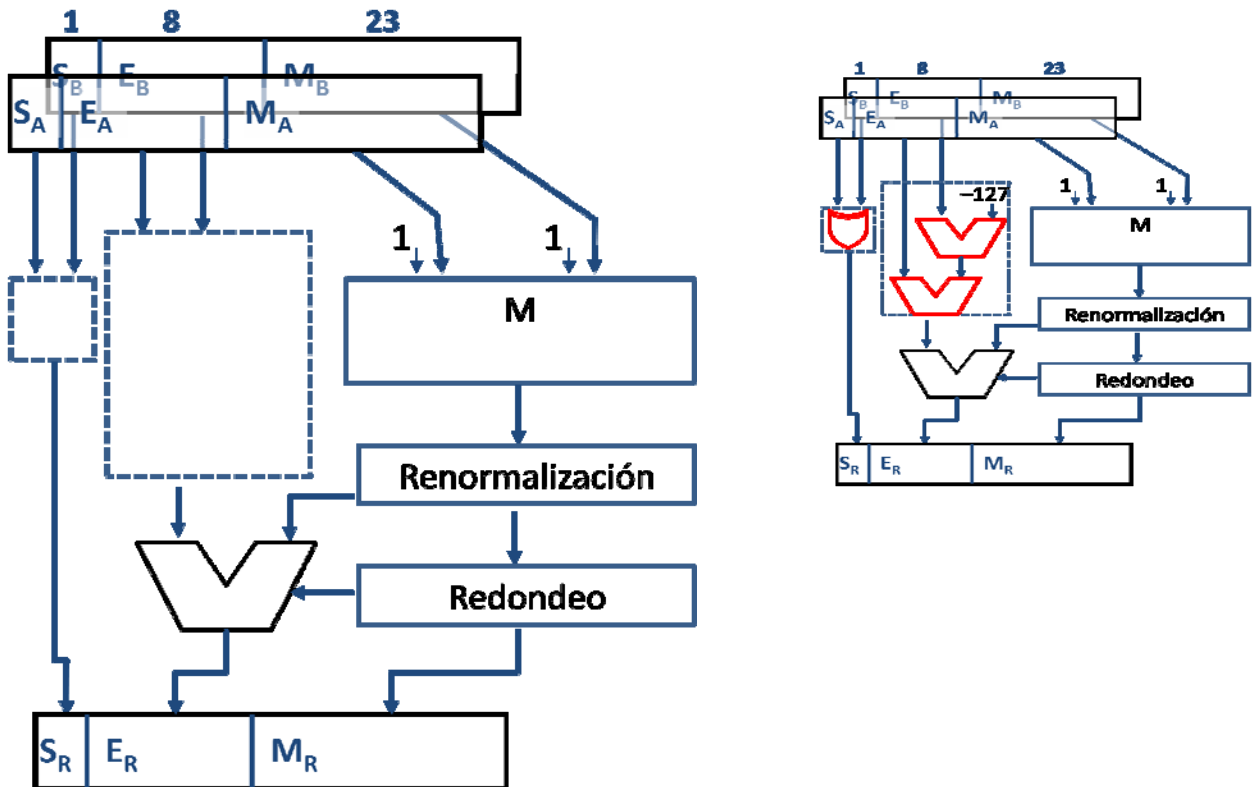
Productividad (CSA 8-8-8-8) = $1/25 \text{ ns} = 40 \text{ MOPS}$

Se observa que es más rápido que el anterior, aunque tiene más FAs y por tanto mayor área de silicio y consumo de potencia.



6 (1 punto)

Dado el siguiente operador multiplicador hardware de coma flotante de la figura, responda las siguientes preguntas.



a) (0.25 puntos) Indique la instrucción del repertorio del MIPS32 a la que corresponde.

MUL.S fd,fs,ft

Multiplicación en coma flotante, simple precisión (note que los registros son de 32 bits)

b) (0.5 puntos) Rellene sobre el dibujo el contenido interno de los cuadros en línea discontinua. Utilice los componentes digitales necesarios. Justifique la respuesta.

$SR = SA \oplus SB \rightarrow$ puerta XOR de 2 entradas

$ER = EA + EB - 127 \rightarrow$ 2 sumadores

Se suele plantear $ER = (EA - 127) + EB$, para evitar posibles situaciones de overflow al hacer $EA + EB$

c) (0.25 puntos) Indique la función del componente M.

Multiplica las mantisas con el bit implícito añadido: $1MA \times 1MB$

Se trata de un multiplicador de enteros sin signo

7 (3 puntos) Considere un chip de memoria SDRAM DDR2 de 8-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 400 MHz.

a. **(0.25 puntos)** Complete la tabla anotando el número de ciclos correspondientes a cada parámetro.

$$t_c = 1/f = 1/400 \text{ MHz} = 2.5 \text{ ns}$$

Parámetro	ns	Ciclos
T_{RCD}	12	$\lfloor 12/2.5 \rfloor = 5$
T_{RAS}	20	$\lfloor 20/2.5 \rfloor = 8$
T_{RC}	50	$\lfloor 50/2.5 \rfloor = 20$
T_{RP}	10	$\lfloor 10/2.5 \rfloor = 4$
CL	—	3

b. **(0.5 puntos)** Dibuje el cronograma de una operación de lectura de dos bloques de 8 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				col2																	
Datos										W0W1	W2W3	W4W5	W6W7	Y0Y1	Y2Y3	Y4Y5	Y6Y7										

c. **(0.25 puntos)** Calcule el ancho de banda del chip.

$$8 \text{ bits} * 400 \text{ MHz} * 2 = 1 \text{ B} * 400 \text{ MHz} * 2 = 800 \text{ MBps}$$

d. **(0.25 puntos)** Calcule el tiempo de acceso de lectura.

$$t_{\text{RCD}} + \text{CL} = 5 + 3 \text{ ciclos} = 8 \text{ ciclos}; 8 \text{ ciclos} * 2.5 \text{ ns} = 20 \text{ ns}$$

e. **(0.25 puntos)** Considere ahora un chip de memoria con parámetros temporales idénticos, pero con tecnología DDR4. Calcule el ancho de banda y el tiempo de acceso de lectura del chip. Indique que ventajas aporta DDR4 frente a DDR2

Para la misma frecuencia, el ancho de banda y el tiempo de acceso es el mismo

La ventaja es que el chip DDR4 podría funcionar a una frecuencia máxima superior a la del DDR2

- f. **(0.25 puntos)** Suponga que un módulo DIMM M está formado por una fila de 8 chips como el anterior. Indique el ancho de banda del módulo para la misma frecuencia.

$$64 \text{ bits} * 400 \text{ MHz} * 2 = 8 \text{ B} * 400 \text{ MHz} * 2 = 6400 \text{ MBps} = 6.4 \text{ GBps}$$

- g. **(0.5 puntos)** Suponga q el módulo M tiene una capacidad de 1GB. Si cada chip contiene 2 bancos de memoria, y cada banco 2K filas, ¿Cuántas columnas tiene cada banco?

$$1\text{GB}/8 = 128\text{MB} = \text{capacidad de cada chip} = 128\text{M} \times 8\text{bits}$$

$$128\text{Mpalabras} = 2^{27} \text{ palabras} = B \times F \times C = 2 \times 2^{11} \times C$$

$$C = 2^{15} \text{ columnas} = 32\text{K columnas}$$

- h. **(0.25 puntos)** Especifique el número de líneas de dirección (banco +filas/columnas) de los chips del módulo M.

Banco: 1 línea, b= 1

Fila/columna: al estar multiplexadas, haría falta $\max(f,c) = \max(11,15) = 15$ líneas

En total, 16 líneas de direcciones

- i. **(0.5 puntos)** Suponga que el módulo M se instala en la placa base de un procesador de 64 bits, que tiene registros de direcciones (dirección efectiva) de 36 bits. El módulo se ubica a partir de la dirección 0x200000000. Indique la dirección final del módulo en el mapa de memoria, y la función de selección a nivel bajo.

$$1\text{GB} = 2^{30}\text{B} \rightarrow \text{el módulo ocupa 30 bits en la dirección efectiva } A_{29} \dots A_0$$

$$\text{Dirección inicial} = A_{35} A_{34} A_{33} A_{32} A_{31} A_{30} A_{29} \dots A_0 = 001000 0\dots 0$$

$$\text{Dirección final} = A_{35} A_{34} A_{33} A_{32} A_{31} A_{30} A_{29} \dots A_0 = 001000 1\dots 1 = 23\text{FFFFFFF}$$

$$\text{Selección} = /Cs = A_{35} + A_{34} + A_{33}^* + A_{32} + A_{31} + A_{30}$$