

Apellidos y Nombre

DNI

Grupo

NOTA IMPORTANTE: puesto que el uso de la calculadora está permitido durante este examen, todos los cálculos deberán llegar hasta su valor final y deberán incluir las unidades en que se expresan

- 1 (2,2 puntos)** La ruta de los datos monociclo del procesador MIPS R2000, que se muestra en la Figura 1, es la que se ha estudiado en clase.

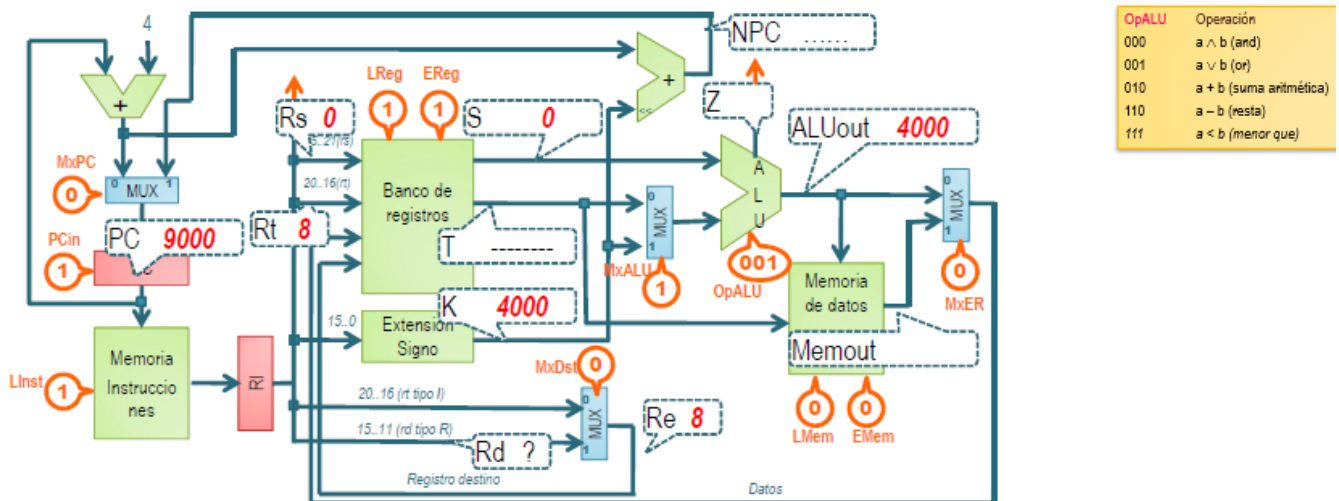


Figura 1. Ruta de los datos monociclo vista en tema 1

- a) **(0,5 puntos)** Tras evaluar la información que circula por la ruta de los datos y las señales de control, responda ¿cuál sería la instrucción que se está ejecutando en el momento actual (indique la instrucción completa, ejemplo lw \$5,12(\$3))?

ori \$8, \$0, 4000

- b) **(0,7 puntos)** Sobre dicha ruta de los datos se han ejecutado las siguientes instrucciones (el valor entre paréntesis de la izquierda es la posición de memoria donde se encuentra almacenada cada instrucción):

[0x0400 0010] ori \$10, \$0, 0x1000

[0x0400 0014] sw \$10, 0(\$12)

Complete los huecos de las tablas, correspondientes a las señales de la anterior ruta cuando se está ejecutando la instrucción **sw**. Marque con un interrogante los valores que sean desconocidos.

Señales	Valor
EReg	0
LReg	1
Rs	12
Rt	10
T	0x1000

Señales	Valor
MxDst	¿?
MxER	¿?
MxALU	1
MxPC	0
NPC	¿?

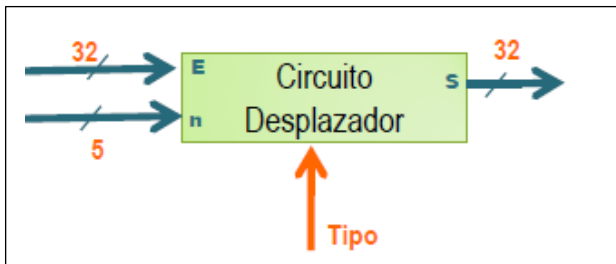
Tabla 1 Tablas valor señales ejecución sw \$10,0(\$12). Apartado 1.b

- c) **(1 punto)** Se desea incluir las instrucciones de *desplazamiento lógico y aritmético* al repertorio de instrucciones del MIPS estudiado en la asignatura. Dichas instrucciones realizan el desplazamiento a izquierda (*sll*) y derecha (*srl* y *sra*) de un registro RT, una cantidad de bits indicada en el campo desp5, dejando el resultado de la operación en el registro destino RD. La diferencia entre *srl* y *sra* radica en que en *srl* entran ceros desde el bit de mayor peso al desplazar y en *sra* se replica el bit de signo.

COP	RS	RT	RD	desp5	Función
000000	00000	00100	00010	00101	000000

Figura 2 Codificación *sll* \$2,\$4,5 de tipo R # \$2 = \$4 * 2⁵ (desplaza cinco posiciones a la izquierda, introduciendo cero)

- o El operador que se propone para ello es el siguiente:

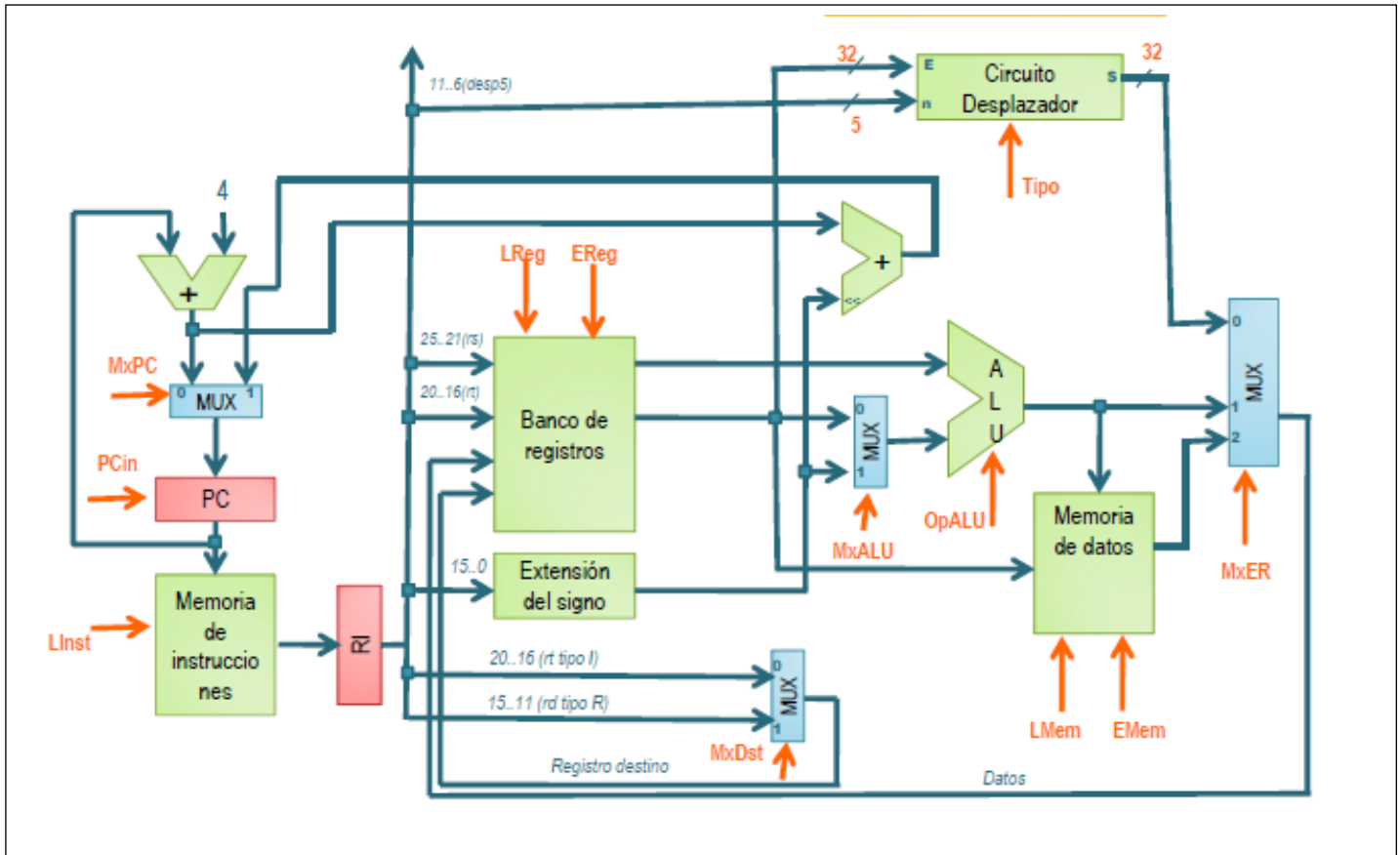


Tipo = 00 $S = E \ll n$ ($E * 2^n$) (SLL)

Tipo = 01 $S = E \gg n$ ($E/2^n$) (SRL)

Tipo = 10 $S = E \gg n$ ($E/2^n$) (SRA)

- o Dibuje sobre la ruta de los datos el lugar dónde se pondría dicho operador e indique si hiciera falta modificar/añadir algún multiplexor.



- o Rellene la tabla con las señales de control necesarias para ejecutar las siguientes instrucciones (en binario).

Instrucción	Form	EReg	MxER	MxDst	OpALU	MxALU	Emem	Lmem	MxPC	Tipo
sll rd,rt,desp5	R	1	00	1	xxx	xxx	0	0	0	00
sra rd,rt,desp5	R	1	00	1	xxx	xxx	0	0	0	10

Tabla 2 Tabla valor señales apartado 1.c

2 (1,3 punto) En el procesador segmentado en cinco etapas, semejante al estudiado en la asignatura, se va a ejecutar el siguiente fragmento de código en ensamblador del MIPS R2000. Asuma que los conflictos por dependencias de datos y control se solucionan mediante la inserción de ciclos de parada y latencia de salto de dos ciclos.

```

(1) Media: beqz $a1,fin
(2)         or $t0,$0,$a1
(3)         or $t1,$0,$0
(4) for1:   lw $t2,0($a0)
(5)         add $t1,$t1,$t2
(6)         addi $a0,$a0,4
(7)         addi $t0,$t0,-1
(8)         bnez $t0,for1
(9)         div $t1,$a1
(10)        mflo $v0
(11) fin:   jr $ra

```

	Registro	instrucción en que se escribe	instrucción en que se lee
Riesgo	\$t1	(3)	(5)
Riesgo	\$t2	(4)	(5)
Riesgo	\$t0	(7)	(8)
Riesgo			
Riesgo			

Tabla 3. Conflictos por dependencias de datos

- a) **(0,3 puntos)** Haciendo uso del número de instrucción del fragmento de código, indique los conflictos por dependencias de datos usando la Tabla 3 (no tiene por qué haber tantos conflictos como filas en la tabla)
- b) **(0,4 puntos)** Rellene la siguiente tabla asumiendo que \$a1 = 100 (justifique las respuestas)

Número de Instrucciones ejecutadas (I)	$3 + (5 * 100) + 3 = 506$ instrucciones
Número de ciclos de parada (P)	$(2_{add} + 2_{bnez}) * 100 \text{ veces} + 2_{beqz} + 2_{jr} + 2_{bnez} * 100 \text{ veces} = 604$
Número de ciclos de parada que se deben al control	$2_{beqz} + 2_{jr} + 2_{bnez} * 100 \text{ veces} = 204$
Número de ciclos totales de ejecución (T)	$I + P + 4 = 506 + 604 + 4 = 1114$ ciclos
CPI	$1 + P / I = 1 + 604/506 = 2.19$

- c) (0,6 puntos) Rellene los 19 primeros ciclos del diagrama de ejecución del código anterior, indicando qué instrucción entra y las etapas en las que se encuentra en cada ciclo:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
(1) beqz \$a1,fin	LI	DI	EX	M	ER														
(2) or \$t0,\$0,\$a1			*	*	LI	DI	EX	M	ER										
(3) or \$t1,\$0,\$0					LI	DI	EX	M	ER										
(4) lw \$t2,0(\$a0)						LI	DI	EX	M	ER									
(5) add \$t1,\$t1,\$t2							LI	*	*	DI	EX	M	ER						
(6) addi \$a0,\$a0,2										LI	DI	EX	M	ER					
(7) addi \$t0,\$t0,-1											LI	DI	EX	M	ER				
(8) bnez \$t0,for1											LI	*	*	DI	EX	M	ER		
(4) lw \$t2,0(\$a0)															*	*	LI	DI	EX

- 3 (1 punto) Considérese la ruta de datos segmentada de cinco etapas (LI, DI, EX, M, ER) estudiada en la asignatura y supóngase que las etapas tienen los siguientes retardos: 30 ns las memorias, 30 ns el banco de registros para lectura y escritura, y 30 ns la unidad aritmético-lógica. Los registros de segmentación tienen un retardo de 5 ns. El resto de unidades funcionales tienen un tiempo despreciable.

- a) (0,5 puntos) ¿Cuál es la aceleración máxima conseguida respecto a la versión no segmentada? Justifique su respuesta

$$T_s = \max \{30,30,30,30,30\} + 5 = 35 \text{ ns}$$

$$T_{NS} = 30 + 30 + 30 + 30 + 30 = 150 \text{ ns}$$

$$S = \frac{T_{NS}}{T_s} = \frac{150}{35} = 4,3$$

- b) (0,5 puntos) ¿Cuál sería la productividad real (MIPS) obtenida por el código de la pregunta 2 en la ruta segmentada de este ejercicio?

$$\chi_{REAL}(506) = \frac{1000 \cdot 506 \text{ instrucciones}}{1114 \text{ ciclos} \times 35} \text{ MIPS} = 12,98 \text{ MIPS}$$

- 4 (1 punto) Considere el circuito de la figura 3, que combina un sumador FA, una puerta NOT y un multiplexor. Asuma que todas las puertas lógicas tienen un retardo T y que el circuito FA es como el estudiado en clase de modo que el acarreo tiene un retardo de 2T y la suma de 3T (desde la llegada de la entrada más lenta).

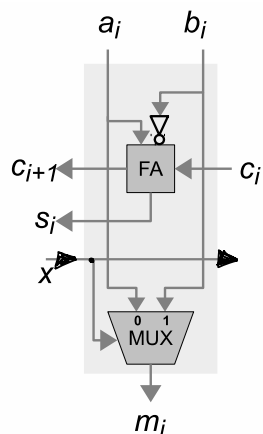


Figura 3. Combinación de un FA y una puerta NOT

Con ese circuito como bloque básico, se puede construir el operador \max de 32 bits de la figura 4, al que se le aplican dos operandos A y B y devuelve el mayor $M = \max(A, B)$. Note que el operador hace la resta $A - B$ y que el bit de signo resultante (s_{31}) lo utilizan los multiplexores para seleccionar entre A y B (los valores representan enteros en complemento a dos).

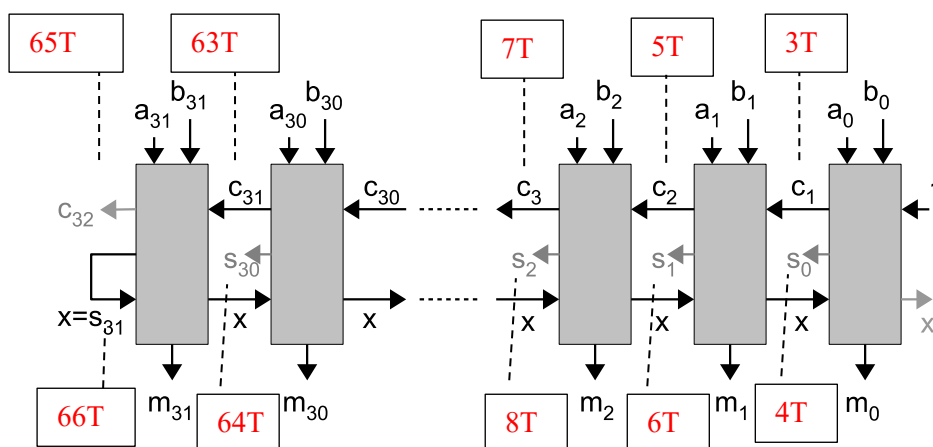


Figura 4: operador \max , el interior de cada elemento se muestra en la Figura 3.

- a) (0,5 puntos) Si las entradas a_i , b_i y c_0 llegan simultáneamente al circuito de la figura 4, ¿cuáles son los retardos de las salidas c_{i+1} y s_i ? Indíquelos sobre el dibujo dentro de los recuadros correspondientes.

$$\text{Retardo de } c_i = \max\{a_i, b_i, c_i\} + 2T \quad \text{Retardo } s_i = \max\{a_i, b_i, c_i\} + 3T$$

En el primer bit la entrada más lenta es b_i (que lleva una NOT adicional dentro del circuito), pero en los demás bits la entrada más lenta es la c_i (que llega del elemento anterior)

- b) (0,5 puntos) Asumiendo que un multiplexor tiene un retardo de 3T y el tiempo T es 1 ns. Indique cuál sería el retardo de este circuito y su productividad máxima en MOPS.

$$\text{Retardo } s_{31}: 66T + T_{\text{mux}} = 69T = 69 \text{ ns}$$

$$\text{Productividad} = 1 \text{ operación} / 69 \text{ ns} \approx 14,5 \text{ MOPS}$$

- 5 (1 punto)** Se dispone de un circuito multiplicador secuencial con **recodificación por parejas de bits** para números enteros de 16 bits tal y como se muestra en la figura 5. Supóngase que el retardo de las diferentes operaciones involucradas en la multiplicación es el que se indica en la tabla posterior:

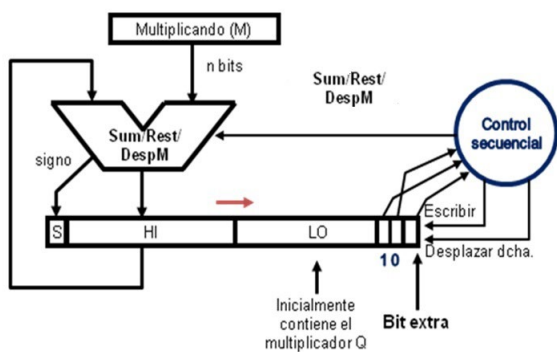


Figura 5

Inicializar registros y circuito de control	4 ns
Inspeccionar q_{i+1} q_i y q_{i-1}	1 ns
Sumar	10 ns
Restar	12 ns
Desplazar M 1 bit (Calcular $2M$)	2 ns
Desplazar S-HI-LO-X 2 bits	4 ns
Escribir registro HI	2 ns
Evaluar el número de ciclo actual	4 ns

- a) **(0,25 puntos)** ¿Cuál sería la productividad que podría alcanzar dicho multiplicador? Asuma que la inicialización de los registros se hace en un ciclo independiente.

$$T_{\text{ciclo}} = \text{Inspección} + 2M + \text{Restar} + \text{Escribir HI} + \text{Desplazar} + \text{Evaluar fin} = 1 + 2 + 12 + 2 + 4 + 4 = 25 \text{ ns}$$

$$T_{\text{op}} = (16/2 + 1) * T_{\text{ciclo}} = 9 * 25 \text{ ns} = 225 \text{ ns}$$

$$\chi = 1 / T_{\text{op}} = 1000/225 \text{ MOPS} = 4,4$$

- b) **(0,75 puntos)** Rellene la tabla siguiente con los ciclos y acciones requeridas para realizar la multiplicación de M y Q siendo M= 54 (0000 0000 0011 0110_{c2}) y Q = -54 (1111 1111 1100 1010_{c2})

$$54 = 0000 \ 0000 \ 0011 \ 0110 \text{ y } Q = -54 = 1111 \ 1111 \ 1100 \ 1010 \text{ luego}$$

$$Q_{\text{Booth}} = 1111 \ 1111 \ 0 \ -1 \ 0 \ +1 \ -1 \ +1 \ -1 \ 0 \text{ y } Q_{\text{pares}} = 0000 \ -1 \ +1 \ -1 \ -2$$

Ciclo	Acción
1	HI = HI -2M
	Desplazar S-HI-LO dos veces
2	HI = HI -M
	Desplazar S-HI-LO dos veces
3	HI = HI +M
	Desplazar S-HI-LO dos veces
4	HI = HI -M
	Desplazar S-HI-LO dos veces

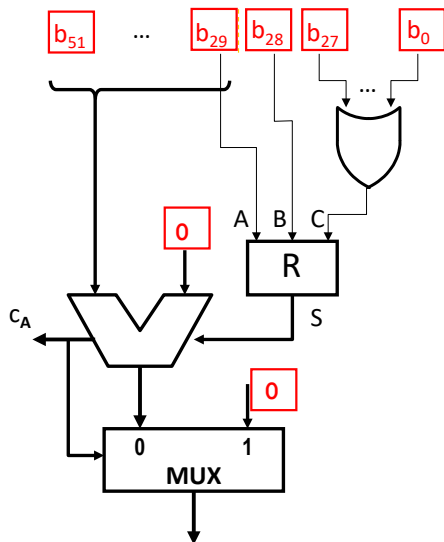
Ciclo	Acción
5	No hacer nada
	Desplazar S-HI-LO dos veces
6	No hacer nada
	Desplazar S-HI-LO dos veces
7	No hacer nada
	Desplazar S-HI-LO dos veces
8	No hacer nada
	Desplazar S-HI-LO dos veces

6 (1 punto) Con relación al operador en coma flotante que da soporte a la instrucción del MIPS *cvt.s.d f_destino, f_origen* (Simple Precisión: signo 1 bit; exponente 8 bits; mantisa 23 bits. Doble Precisión: signo 1 bit; exponente 11 bits; mantisa 52 bits)

a) (0.5 puntos) Indique cómo se deben calcular el signo, mantisa y exponente

Bit de signo	$S_s = S_d$
Exponente	<p>Cambiar de 11 bits en exceso 1023 a 8 bits en exceso 127:</p> $E_s = (E_d - 1023) + 127 = E_d - 896$
Mantisa	Pasar de 52 a 23 bits. Eliminar 29 bits por la derecha y redondear

b) **(0.5 puntos)** La figura 6 muestra el circuito de redondeo al entero más próximo (sesgado al par) como parte de este operador. Inserte los valores que faltan que corresponden a los bits de la mantisa doble precisión y los valores del recuadro de la entrada del sumador y multiplexor. Además, complete la tabla de verdad (Tabla 4) del circuito “R” con los valores que se deben obtener en la salida S a partir de las entradas A, B, C



A	B	C	S
x	0	x	0
0	1	0	0
1	1	0	1
x	1	1	1

Tabla 4. Tabla de verdad circuito R

Figura 6: Circuito de redondeo al entero más próximo (sesgado al par)

7 (2,5 puntos) Un computador dispone de un procesador de 64 bits que direcciona hasta 8 GB, se le conectan los módulos de memoria: M1 de 2GB en las direcciones inferiores del mapa de memoria, M2 en el rango 0x1A0000000 - 0x1BFFFFFFF y finalmente M3 de 1GB ubicado en las direcciones superiores del mapa de memoria..

a) **(0,2 puntos)** ¿Cuántos bits de dirección lógica tendrá el procesador indicado y cómo se denominarán?

Para direccionar los 8 GB se necesitarán 33 bits ($2^{33} = 8G$)

$A_{32} \dots A_0$

b) **(0,4 puntos)** ¿Cuál será la dirección inicial y final del módulo M3 expresadas en hexadecimal?

1 GB = 2^{30} B → Se precisan 30 bits para direccionarlo

Dirección final: 0x1FFFFFFF Dirección Inicial: 0x1C000000

c) **(0,4 puntos)** ¿Cuál será la función para seleccionar M3 suponiendo que se activa por nivel alto?

Si CS es activo por nivel alto: $CS_M1 = A_{32} \cdot A_{31} \cdot A_{30}$

d) **(0,3 puntos)** ¿Qué capacidad en bytes tendrá el módulo M2?

Para direccionar M2 se precisan 29 líneas: 1101XXX ... XXX → $2^{29} = 512$ MB

El módulo M1 de 2GB está compuesto por una única fila de chips de memoria del tipo SDRAM-DDR3-1600 con ancho de palabra 8 bits y conectados a un reloj de 800MHz.

e) **(0,2 puntos)** ¿Cuántos chips tendrá M1 y cuál será la capacidad de cada uno de los chips?

Harán falta: $64 \text{ bits ancho palabra procesador} / 8 \text{ bits ancho palabra chip} = 8 \text{ chips}$

M1 tiene una capacidad de 2GB y está compuesto por 8 chips

$2GB / 8 \text{ chips} = 256 \text{ MB}$ es la capacidad de cada chip → $256 \text{ M} \times 8 \text{ bits}$

f) **(0,4 puntos)** ¿Cuántas líneas DMQi (selección de octeto) tendrá M1? Justificar la respuesta

8 líneas, una por cada byte: BE0*, BE1*, BE2*, BE3*, BE4*, BE5*, BE6*, BE7*

Cada uno de los chips de M1 tiene las especificaciones temporales siguientes: CL - t_{RCD} - t_{RP} - t_{RAS} : 4-2-6-8. Además, está organizado internamente en 16 bancos de 64 K filas, siendo el tamaño de bloque de 8 palabras.

- g) **(0,6 puntos)** Complete la siguiente tabla con la información del chip. Utilice, donde sea apropiado, los prefijos habituales, como por ejemplo “18 K palabras” o “2 MB”

Número de palabras que contiene el chip	256 M palabras
Número de bits para seleccionar banco	$\log_2 \text{Num_Bancos} = \log_2 16 = 4 \text{ bits}$
Número de bits para seleccionar una fila	$64 \text{ K} = 2^{16} \rightarrow 16 \text{ bits}$
Número de bits para seleccionar un bloque	$\text{Num_bloques} = 256 \text{ M palabras} / 8 = 32 \text{ M bloques}$ $\log_2 32768 = 15 \text{ bits}$
Capacidad en bytes de cada banco	$256 \text{ MB} / 16 \text{ bancos} = 16 \text{ MB cada banco}$
Capacidad en bytes de cada fila de un banco	$16 \text{ MB cada banco} / 64 \text{ K filas} = 256 \text{ B}$