

# EJERCICIOS

## Actividad

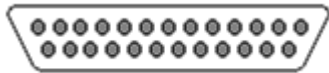
---

Abre el PC y encuentra e identifica los siguientes elementos

### CONECTORES EXTERNOS



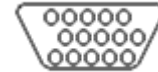
a) PS/2



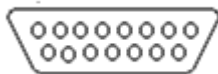
b) puerto paralelo



c) puertos serie



d) Puerto VGA



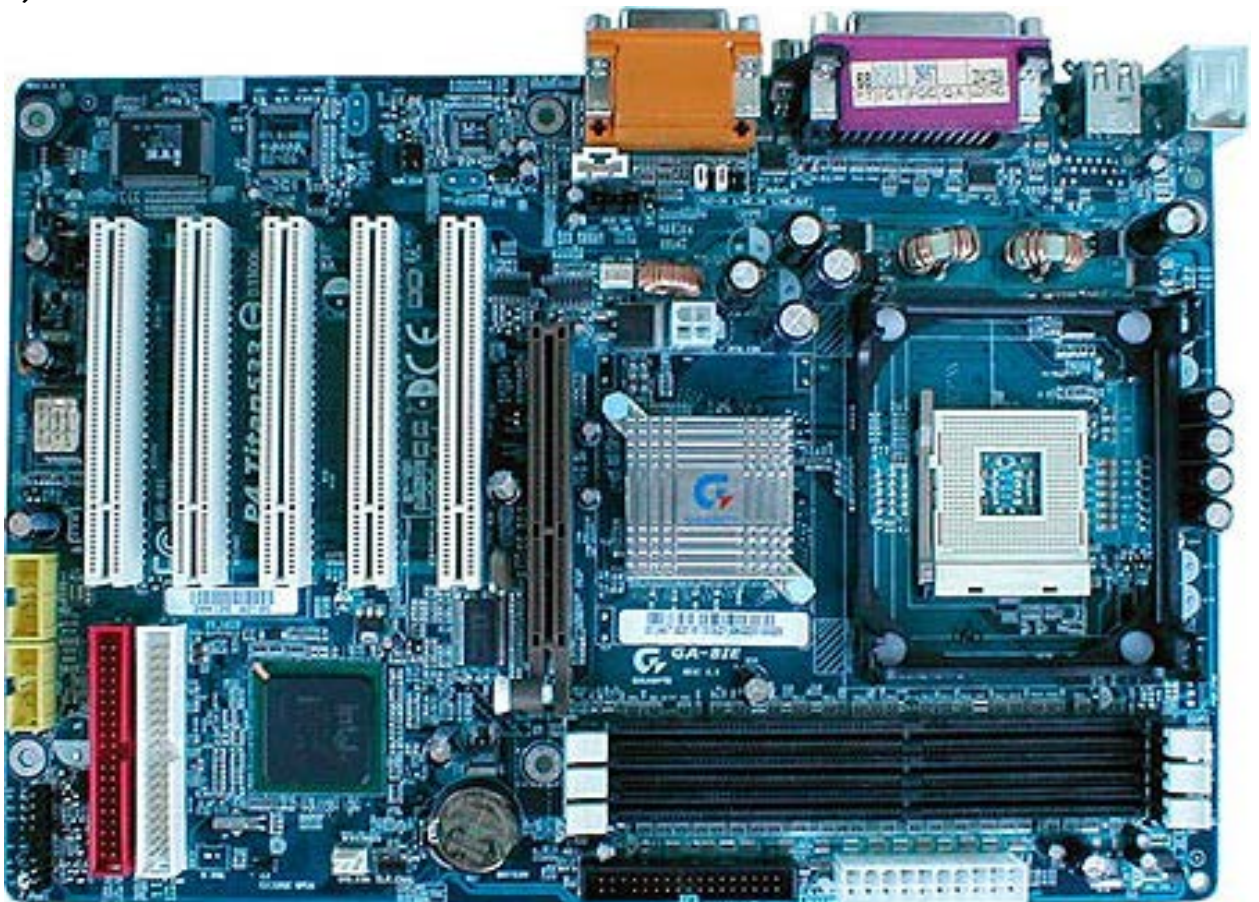
e) puerto de juegos



f) conector DVI-I

g) USB

h) PLACA BASE



## TARJETAS DE EXPANSIÓN

- i) Tarjeta gráfica
- j) Tarjeta de red

## CONECTORES Y SLOTS INTERNOS

- k) IDE (para conectar el CD, HD,...)
- l) PCI (para las tarjetas de expansión)
- m) conectores para la memoria RAM
- n) zócalo para el microprocesador

## DISPOSITIVOS

- ñ) Disco duro (memoria ROM)
- o) Lector/Grabador de CD/DVD
- p) Memoria RAM
- q) Microprocesador
- r) Pila del reloj del sistema
  
- s) BIOS

## Tema 0

---

### 1. Operaciones en el sistema binario (*ejercicios con solución*):

<b>Suma</b> a) $11 + 11 =$ b) $100 + 10 =$ c) $111 + 11 =$ d) $110 + 100 =$	<b>Producto</b> e) $11 \cdot 1 =$ f) $11 \cdot 11 =$ g) $111 \cdot 101 =$ h) $1011 \cdot 1001 =$
<b>Resta</b> i) $11 - 01 =$ j) $11 - 10 =$ k) $111 - 100 =$ l) $110 - 101 =$ m) $111 - 11 = 011$	<b>División</b> n) $110 : 11 =$ o) $110 : 10 =$

### 2. Conversión de decimal a binario

- a) 15
- b) 59
- c) 0,3125
- d) 31,15

### 3. Conversión de binario a hexadecimal

- a) 1100101001010111
- b) 01101001101

### 4. Conversión de hexadecimal a binario

- a)  $10A4_{16}$
- b)  $CF8E_{16}$
- c)  $9742_{16}$

### 5. Conversión de decimal a hexadecimal

- a)  $650_{10}$
- b)  $4025_{10}$

### 6. ¿Cuál es el peso del 6 en los siguientes números decimales?

- a) 1386

- b) 54,692
- c) 671,920

7. Expresar los siguientes números en potencias de 10

- a) 10 =
- b) 100 =
- c) 10000 =
- d) 1000000 =

8. ¿Cuál es el mayor número que se puede representar con 4 dígitos decimales?

9. Pasar a decimal estos números dados en binario:

- a) 11
- b) 100
- c) 111
- d) 1000
- e) 11101
- f) 11,011

10. ¿Cuál será el mayor valor decimal representable con este número de bits?

- a) 2 bit
- b) 7 bit
- c) 10 bit

11. ¿Cuántos bits necesitamos para poder representar los siguientes números decimales?

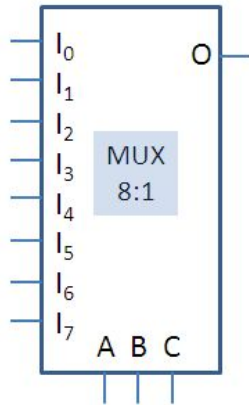
- a) 17
- b) 81
- c) 35
- d) 32

12. Pasar a decimal estos números de diferentes bases:

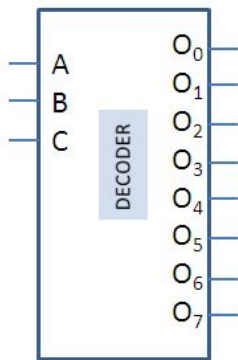
- a)  $E_{16}$
- b)  $B2F8_{16}$
- c)  $2374_8$

13. Convierte los siguientes números decimales a binario con un mínimo de 3 dígitos después de la coma y un error relativo  $< 0,1\%$
- a) 2,08
  - b) 73,625
14. Los siguientes números representan números en coma flotante en el Std. 754, ¿a qué números decimales corresponden?
- a)  $01000010000110110000000000000000_2$
  - b)  $00111011000101101011101110011000_2$
  - c)  $470BBBDD_{16}$
  - d)  $7FD04381_{16}$
15. ¿Cuántos bits necesitamos para aplicar Hamming al siguiente dato? ¿Los bits de qué posiciones llevarán bits de datos y cuáles bits de control?
- a) D3 D2 D1 D0 = 1001
  - b) D6 D5 D4 D3 D2 D1 D0 = 0110111
16. Hamming en transmisión: añade los bits de control necesarios a los datos del ejercicio anterior.
17. Hamming en recepción: comprueba si los datos recibidos (a los que se aplicó Hamming en transmisión) han sufrido algún error, corrigiéndolos en caso afirmativo. ¿Qué puerta lógica utilizarías para corregir un bit erróneo? ¿Qué sucede si 2 bits han sufrido error? ¿Cómo puede detectarse tal situación?
- a) Dato recibido: 1111010
  - b) Dato recibido: 11101000110
  - c) Dato recibido: 1111111
  - d) Dato recibido: 111101011

18. Implementa la siguiente función canónica con un multiplexor de 8 a 1 como el de la figura.  $f(A,B,C)=\Sigma (m_1, m_3, m_4, m_7)$



19. Implementa la misma función con un decodificador de 3 entradas como el de la figura y el número de puertas lógicas de dos entradas necesarias.



20. Simplifica mediante Karnaugh en forma de maxterm y minterm la función de los ejercicios anteriores y dibuja los circuitos resultantes utilizando puertas lógicas de 2 y 3 entradas.

## SOLUCIONES

1) Operaciones en el sistema binario:

### Suma

- a) 110
- b) 110
- c) 1010
- d) 1010

### Producto

- e) 11
- f) 1001
- g) 100011
- h) 1100011

### Resta

- i) 10
- j) 01
- k) 011
- l) 001
- m) 100

### División

- n) 10
- o) 11

2) Conversión de decimal a binario

$$\begin{aligned}15_{10} &= 1111_2 \\59_{10} &= 111011_2 \\0,3125_{10} &= 0,0101_2 \\31,125_{10} &= 11111,001_2\end{aligned}$$

3) Conversión de binario a hexadecimal

$$\begin{aligned}1100101001010111_2 &= CA57_{16} \\01101001101_2 &= 34D_{16}\end{aligned}$$

4) Conversión de hexadecimal a binario

$$\begin{aligned}10A4_{16} &\rightarrow 1000010100100_2 \\CF8E_{16} &\rightarrow 1100111110001110_2 \\9742_{16} &\rightarrow 1001011101000010_2\end{aligned}$$

5) Conversión de decimal a hexadecimal

$$\begin{aligned}650_{10} &= 28A_{16} \\4025_{10} &= FB9_{16}\end{aligned}$$

6) ¿Cuál es el peso del 6 en los siguientes números decimales?

$$1386 \quad (10^0 \rightarrow \text{bit de peso } 0)$$



54,692                      ( $10^{-1} \rightarrow$  bit de peso -1)  
671,920                      ( $10^2 \rightarrow$  bit de peso 2)

7) Expresar los siguientes números en potencias de 10

10 =  $10^1$   
100 =  $10^2$   
10000 =  $10^4$   
1000000 =  $10^6$

8) ¿Cuál es el mayor número que se puede representar con 4 dígitos decimales?

xxxx  $\rightarrow 10^4 - 1 \rightarrow 9999$

9) Pasar a decimal estos números dados en binario:

11 = 3  
100 = 4  
111 = 7  
1000 = 8  
11101 = 29  
11,011 = 3,375

10) ¿Cuál será el mayor valor decimal representable con este número de bits?

2 bit  $\rightarrow 3$   
7 bit  $\rightarrow 127$   
10 bit  $\rightarrow 1023$

11) ¿Cuántos bits necesitamos para poder representar los siguientes números decimales?

$17_{10} \rightarrow 4$  bits  
 $81_{10} \rightarrow 6$  bits  
 $35_{10} \rightarrow 5$  bits  
 $32_{10} \rightarrow 5$  bits

12) Pasar a decimal estos números de diferentes bases:

$E5_{16} = 229_{10}$   
 $B2F8_{16} = 45816_{10}$   
 $2374_8 = 1276_{10}$

13) a) 10,000101    Erel=0,0901%<0,1%  
b) 1001001,101    E=0

14) a)  $38,75_{10}$     b) -0,0023    c) 35768    d) NaN

15) a) m=4; r=3; \_ \_ D3 \_ D2 D1 D0  
b) m=7; r=4; \_ \_ D6\_ D5 D4 D3 \_ D2 D1 D0

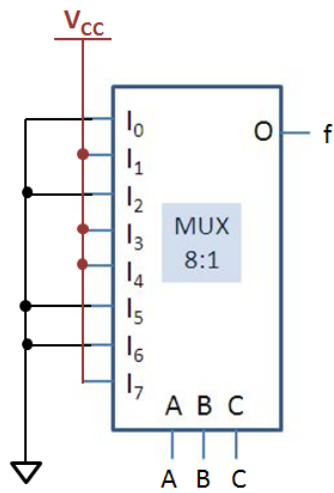
16) a) dato: 1001 → 0011001; b) dato: 0110111 → 1100110111

17) a) error en b2, dato: 1010 b) error en b6, dato: 1110110  
 c) No error, dato: 1111 d) error en b3, dato: 00101

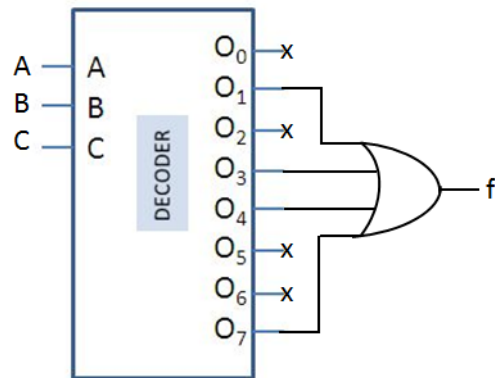
Puede corregirse con la XOR del bit erróneo y la señal de error.

Este método solo puede corregir errores de 1 bit, por lo que aplicar la corrección podría llevar a errores mayores. Un bit de paridad adicional nos ayuda a detectar errores múltiples.

18)



19)



20)  $f = (A+C) \cdot (/B+C) \cdot (/A+B+/C)$   
 $f = /A \cdot C + B \cdot C + A \cdot /B \cdot /C$

## Ejercicios Tema 1

---

### Ruta de datos: Unidad Aritmético Lógica

Diseña una UAL que teniendo como entrada los números A y B de cuatro bits ( $A[A_3, A_2, A_1, A_0]$  y  $B[B_3, B_2, B_1, B_0]$ ), realice las siguientes operaciones:

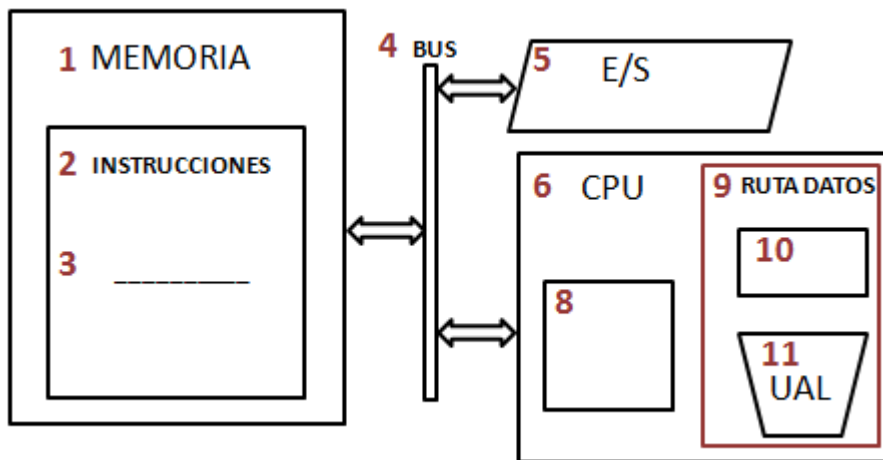
*Aritméticas:* A; incremento de A ( $A+1$ ); suma ( $A+B$ ); resta ( $A-B$ )

*Lógicas:* B; AND ( $A \cdot B$ ); OR ( $A+B$ ); XOR ( $A \oplus B$ )

Dibujar la tabla y el circuito para las operaciones aritméticas, y la tabla y el circuito de una etapa para las operaciones lógicas. ¿Cuántas señales de selección has necesitado? ¿Puedes reducir ese número?

### Von Neumann y Ruta de Datos

1- Rellenar los espacios numerados.



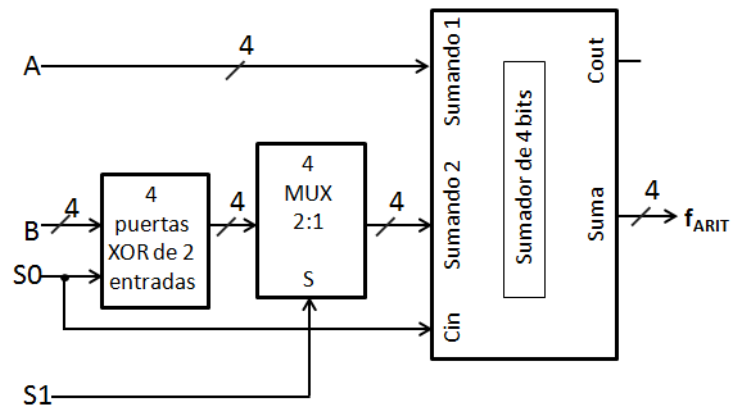
2- Enumerar las principales características de la arquitectura von Neumann.

3- ¿Qué es la palabra de control? ¿Quién la utiliza y para qué?

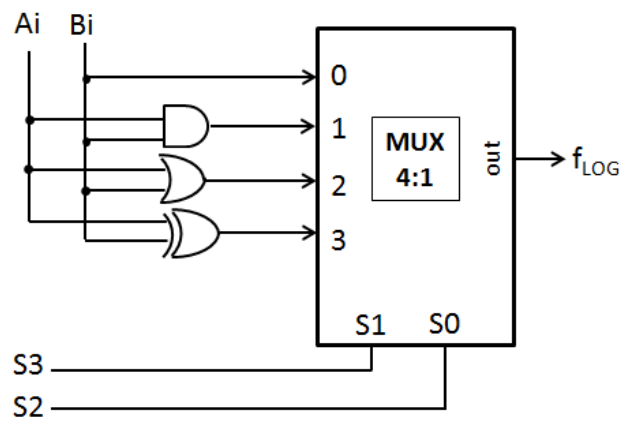
4- ¿Quién genera la palabra de control? ¿A partir de qué información?

## Ruta de datos: Unidad Aritmético Lógica (posible solución)

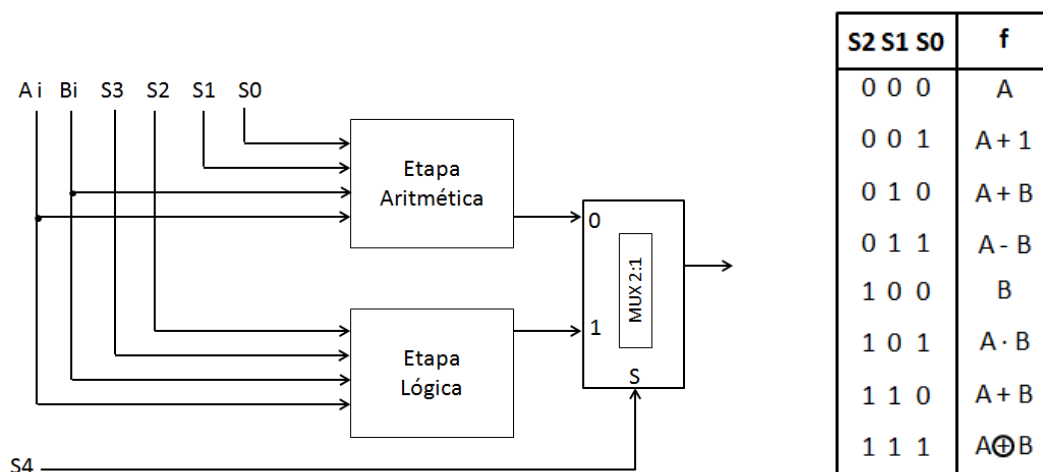
S1	S0	f <sub>ARIT</sub>
0	0	A
0	1	A + 1
1	0	A + B
1	1	A - B



S3	S2	f <sub>LOG</sub>
0	0	B
0	1	A · B
1	0	A + B
1	1	A ⊕ B



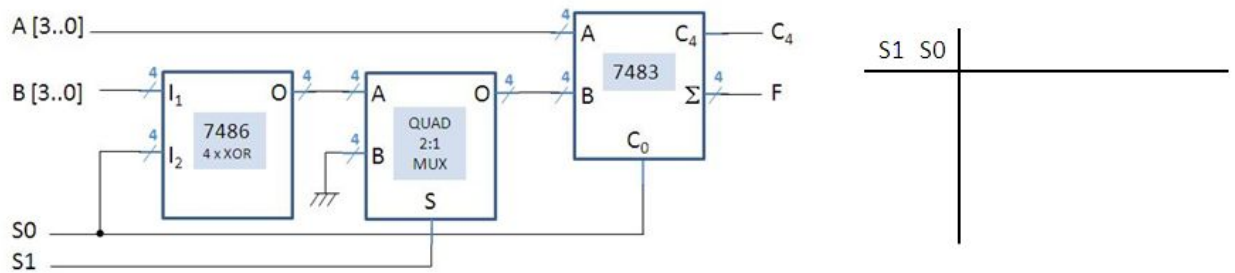
Implementándolo así necesitaría 5 señales de selección: 2 para la etapa aritmética, 2 para la lógica y una quinta para seleccionar entre ambas (imagen de la izquierda). Sin embargo sabemos que para 8 operaciones sería suficiente con 3 señales de selección, es cuestión de reordenarlas (tabla de la derecha).



S2	S1	S0	f
0	0	0	A
0	0	1	A + 1
0	1	0	A + B
0	1	1	A - B
1	0	0	B
1	0	1	A · B
1	1	0	A + B
1	1	1	A ⊕ B

## Unidad aritmético(-lógica)

Identifica y pon en la tabla las funciones que realiza la pequeña unidad aritmética de la figura:

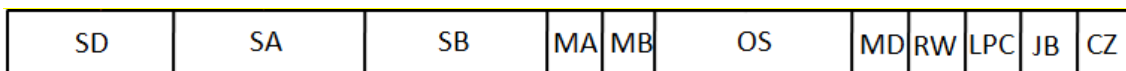


## Unidad aritmético(-lógica) (solución)

S1	S0	Entradas sumador			f
		A	B	Cin	
0	0	A	B	0	A + B
0	1	A	/B	1	A - B
1	0	A	0	0	A
1	1	A	0	1	A + 1

## Ruta de datos. Palabra de control.

- Abajo tenemos el esquema de la palabra de control vista en clase. Cuántos bits tiene y qué indica cada uno de los campos.



SD [     ]:

SA [     ]:

SB [     ]:

MA [   ]:

MB [   ]:

OS [     ]:

MD [   ]:

RW [   ]:

LPC [   ]:

JB [   ]:

CZ [   ]:

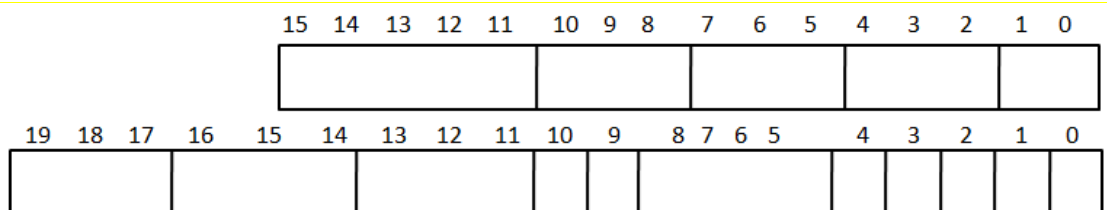
2. Centrándonos solo en la señales de la ruta de datos (obviamos las de control de salto) ¿qué palabra de control generarán las microoperaciones 1, 2 y 3?  
 ¿Qué microoperaciones están en el origen de las palabras de control 4, 5 y 6?

	Microoperación	SD	SA	SB	MA	MB	OS	MD	RW
1	$(R4) \leftarrow sl (R6)$						1001		
2	$(R1) \leftarrow (R0)+2$						0010		
3	$[M] \leftarrow (ACC)$						1100		
4		001	010	011	0	0	0011	0	1
5		111	111	XXX	0	X	0001	0	1
6		XXX	XXX	XXX	0	X	XXXX	1	1

ACC: registro acumulador =R0

### Unidad de control.

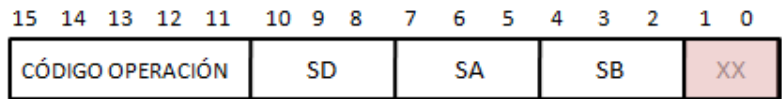
1. De la representación de la instrucción y haciendo uso de las figuras adjuntas (a), (b) y (c), obtened el código binario de la instrucción así como la palabra de control que dará el decodificador cableado para las siguientes instrucciones (*ejercicios con solución*):



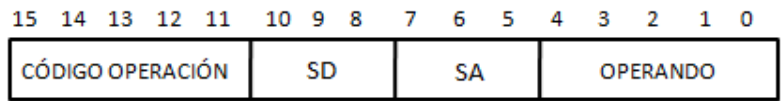
- $(R2) \leftarrow (R5)$
- $(R2) \leftarrow (R1) + (R5)$
- $(R2) \leftarrow 05$
- $(R2) \leftarrow (R1) + 05$

	DESCRIPCIÓN	CÓDIGO DE OPERACIÓN	ENSAMBLADOR
1	$(RD) \leftarrow (RA) - 1$	00000	DCR RD, RA
2	$(RD) \leftarrow (RA) + 1$	00001	INR RD, RA
3	$(RD) \leftarrow (RA) + (RB)$	00010	ADD RD, RA, RB
4	$(RD) \leftarrow (RA) - (RB)$	00011	SUB RD, RA, RB
5	$(RD) \leftarrow / (RA)$	00100	MVN RD, RA
6	$(RD) \leftarrow (RA) \text{ AND } (RB)$	00101	AND RD, RA, RB
7	$(RD) \leftarrow (RA) \text{ OR } (RB)$	00110	OR RD, RA, RB
8	$(RD) \leftarrow (RA) \text{ XOR } (RB)$	00111	XOR RD, RA, RB
9	$(RD) \leftarrow (RB)$	01000	MOV RD, RB
10	$(RD) \leftarrow \text{shift left } (RB)$	01001	SL RD, RB
11	$(RD) \leftarrow \text{shift right } (RB)$	01010	SR RD, RB
12	$(RD) \leftarrow \text{CERO}$	01011	CLR RD
13	$[M] \leftarrow (AC)$	01100	STA
14	$(AC) \leftarrow [M]$	01101	LDA
---		01110	---
---		01111	---
---		10000	---
---		10001	---
15	$(RD) \leftarrow (RA) + OP$	10010	ADI RD, RA, OP
16	$(RD) \leftarrow (RA) - OP$	10011	SUI RD, RA, OP
---		10100	---
17	$(RD) \leftarrow (RA) \text{ AND } OP$	10101	ANI RD, RA, OP
18	$(RD) \leftarrow (RA) \text{ OR } OP$	10110	ORI RD, RA, OP
19	$(RD) \leftarrow (RA) \text{ XOR } OP$	10111	XRI RD, RA, OP
20	$(RD) \leftarrow OP$	11000	MVI RD, OP
21	$(RD) \leftarrow \text{shift left } OP$	11001	SLI RD, OP
22	$(RD) \leftarrow \text{shift right } OP$	11010	SRI RD, OP
---		11011	---
23	$PC \leftarrow \text{ADDRESS}$	11100	JMP
24	$PC \leftarrow \text{ADDRESS } (Z=1)$ $PC \leftarrow PC+1 (Z=0)$	11101	JZ
25	$PC \leftarrow PC + \text{offset}$	11110	BR
26	$PC \leftarrow PC + \text{offset } (Z=1)$ $PC \leftarrow PC+1 (Z=0)$	11111	BRZ

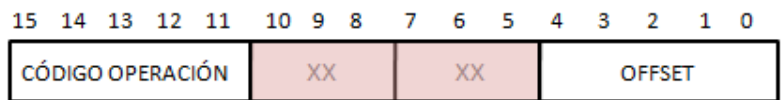
(a) Tabla de instrucciones contempladas por la computadora diseñada



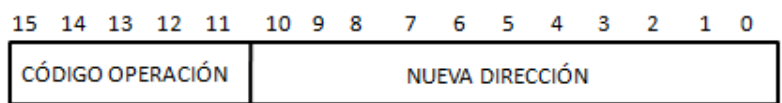
Operación con registros



Operando inmediato

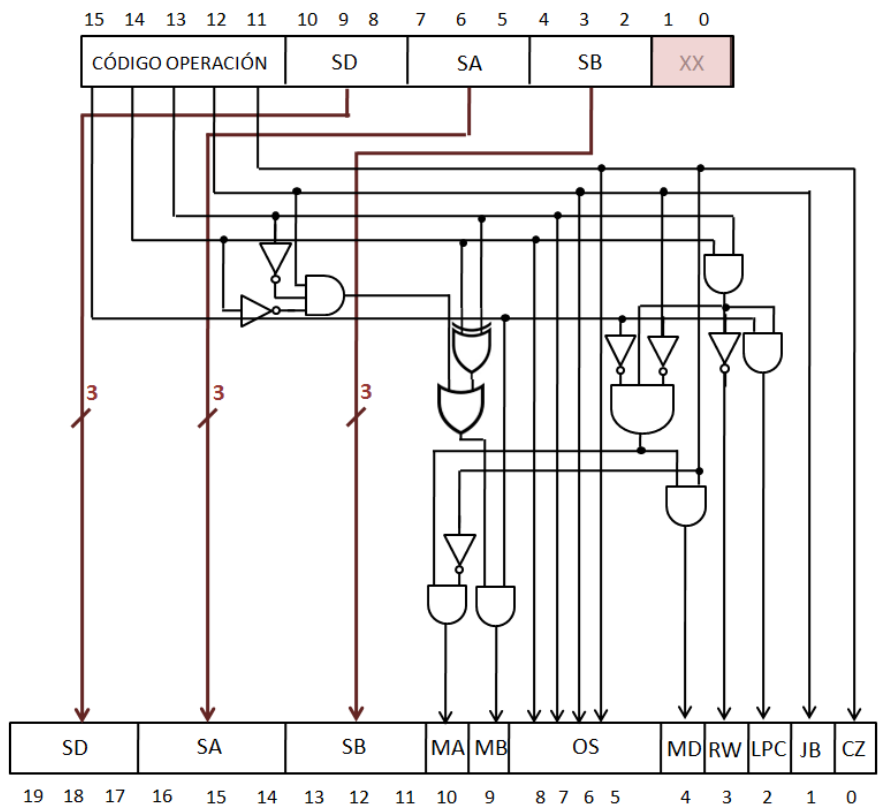


Instrucción de bifurcación



Instrucción de salto

(b) Formatos de instrucción contemplados por la computadora diseñada



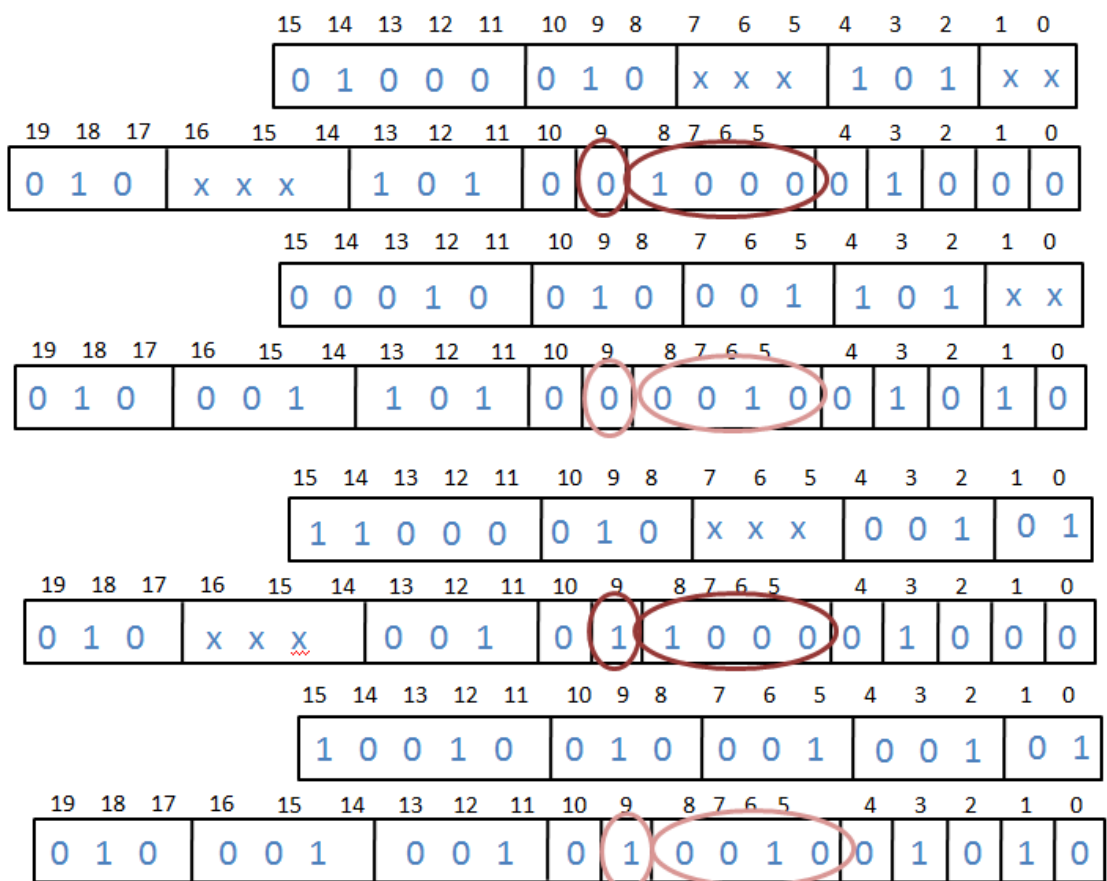
(c) Decodificador cableado



Ruta de datos. Palabra de control (solución del 1. en la parte de teoría)

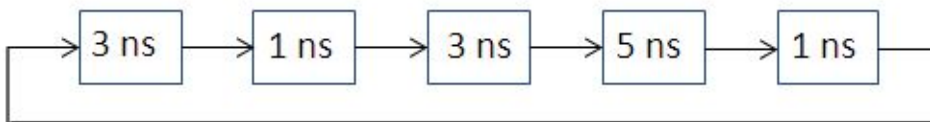
	Microoperación	SD	SA	SB	MA	MB	OS	MD	RW
1	$(R4) \leftarrow sl (R6)$	100	XXX	110	0	0	1001	0	1
2	$(R1) \leftarrow (R0)+2$	001	000	010	0	1	0010	0	1
3	$[M] \leftarrow (ACC)$	XXX	XXX	XXX	1	X	1100	X	0
4	$(R1) \leftarrow (R2)-(R3)$	001	010	011	0	0	0011	0	1
5	$(R1) \leftarrow (R7)+ 1$	111	111	XXX	0	X	0001	0	1
6	$(ACC) \leftarrow [M]$	XXX	XXX	XXX	0	X	XXXX	1	1

Unidad de control (solución)



### Ruta de datos en canalización (pipe-line)

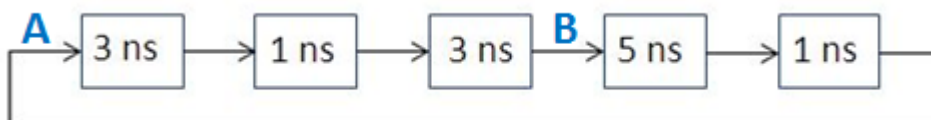
1. Una computadora tiene unos elementos con los retardos indicados en la figura.
  - a. ¿Cuánto tardará en ejecutar 5 instrucciones?
  - b. ¿Cuál será la frecuencia máxima de reloj?
  - c. Se dispone de 2 registros para implementar una ejecución en canalización: ¿dónde los pondrías para obtener la máxima mejora?
  - d. Si cada uno de los nuevos registros añade 1ns, ¿cuánto se tarda ahora en ejecutar una instrucción? ¿Y en ejecutar 5?



2. Una ruta de datos está compuesta por 5 elementos (A-E) que forman un bucle; siendo los tiempos de retardos máximos los siguientes: A (2ns), B (1ns), C (3ns), D (4ns) y E (4ns)
  - a) ¿Cuál es la mayor frecuencia de trabajo?
  - b) Disponemos de 3 registros para organizar la canalización. Divide los 5 elementos en las etapas que consideres más convenientes. ¿Qué frecuencia de trabajo podemos obtener ahora?
  - c) Repite los apartados a) y b) para una canalización de 4 etapas.

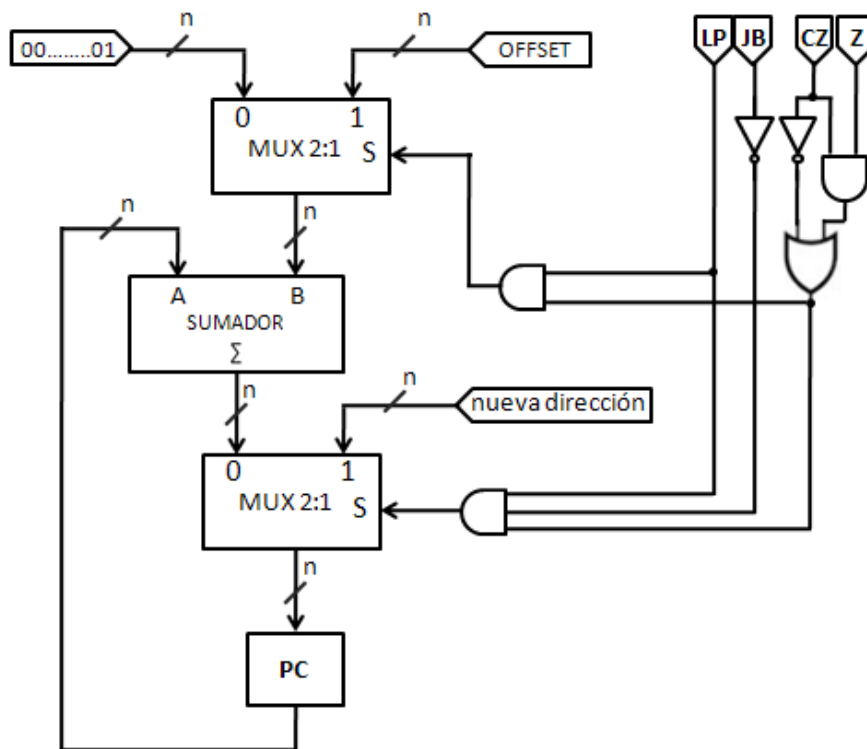
### Ruta de datos en canalización (pipe-line) (solución ejercicio 1)

a) 65 ns    b) 15,38 MHz    c) puntos A y B de la imagen    d) 16 ns. 48 ns.



## Actualización PC

Analiza el circuito de la figura y rellena la tabla .



LPC	JB	CZ	Z	Carga en PC	
0	X	X	X		
1	0	0	X		
1	0	1	0		
1	0	1	1		
1	1	0	X		
1	1	1	0		
1	1	1	1		

## Actualización PC (solución)

LPC	JB	CZ	Z	Carga en PC	
0	X	X	X	PC + 1	<i>No se carga nada especial en el acumulador</i>
1	0	0	X	Nueva dirección	<i>Salto</i>
1	0	1	0	PC + 1	<i>Bifurcación condicionada a Z (flag de cero)</i>
1	0	1	1	PC + offset	<i>Bifurcación condicionada a Z (flag de cero)</i>
1	1	0	X	PC + offset	<i>Bifurcación</i>
1	1	1	0	PC + 1	<i>Bifurcación condicionada a Z (flag de cero)</i>
1	1	1	1	PC + offset	<i>Bifurcación condicionada a Z (flag de cero)</i>

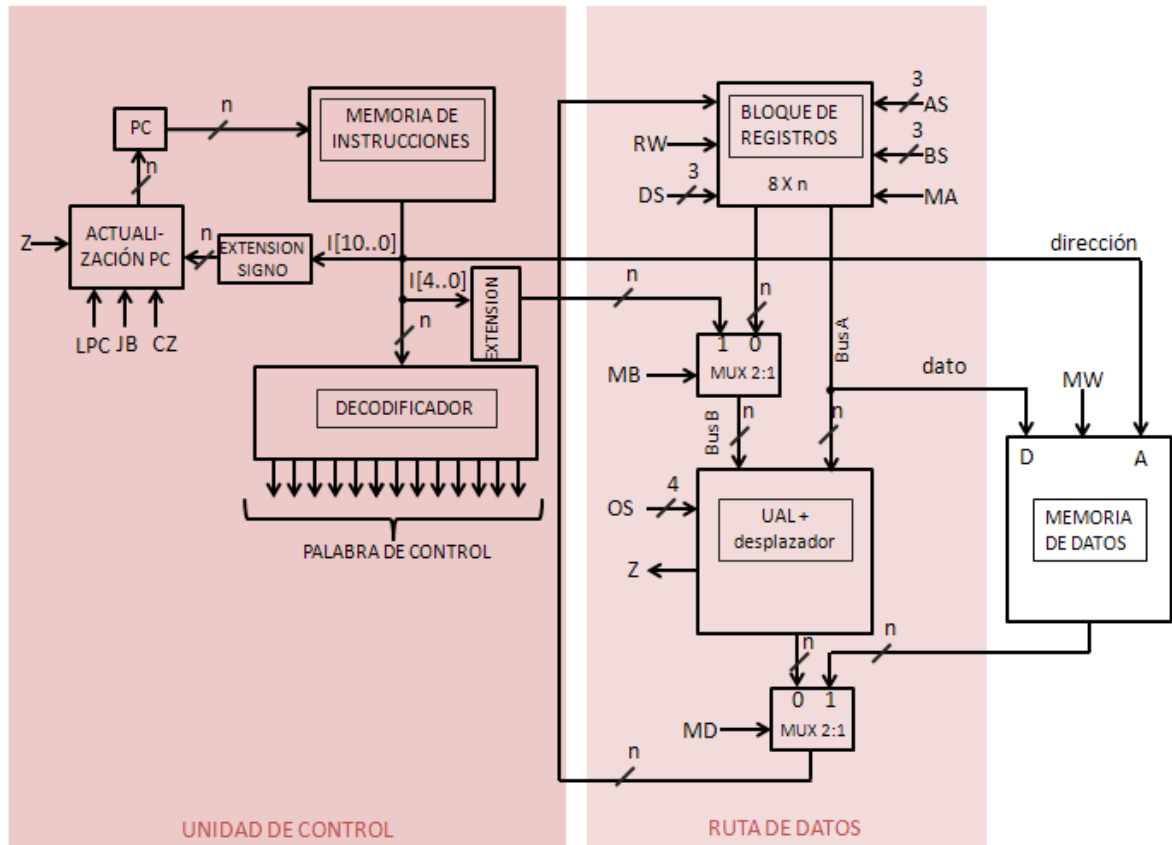
Si LPC=0 no se carga nada especial en el acumulador, sigue la secuencia normal (siguiente instrucción). Si LPC=1 y CZ=1 realizará una bifurcación condicionada a que el resultado de la operación anterior fuera o no cero. En caso de que LPC=1 y CZ=0, dependiendo de JB realizará JB=0 un salto o JB=1 una bifurcación.

## Secuenciación y control

Análisis de las posibles unidades de control y sus diferencias.

1. ¿Para qué se necesitan las señales de selección presentes en la Ruta de Datos?
2. Las señales de selección que se ven en las figuras, ¿de dónde vienen? ¿Quién las genera y a raíz de qué información?
3. ¿Cuál es el cometido de la Unidad de Control?
4. ¿Cuáles son las diferencias fundamentales entre una unidad de control cableada y una microprogramada? Explica el funcionamiento básico de cada una de ellas.
5. ¿Por qué en la unidad de control de ciclo único es necesario que las instrucciones y los datos estén en diferentes memorias?
6. ¿Por qué la palabra de control será más larga para la unidad de control de ciclo múltiple?
7. ¿Qué se hace en el rectángulo que indica "EXTENSIÓN"? ¿Cuándo se utiliza? ¿Por qué es necesario? ¿Y el bloque "EXTENSIÓN DE SIGNO"?
8. Si el código de operación es de 5 bits y el control es microprogramado de ciclo múltiple, ¿es suficiente con una memoria de microprograma de  $2^5$  posiciones? Razona la respuesta.
9. Si tras el diseño de la Ruta de Datos y del control, queremos añadir una nueva operación, ¿cuáles son los pasos a dar en un caso y en el otro?

## Control cableado, ciclo sencillo



## Secuenciación y control (solución)

1. Para dirigir las operaciones (aritméticas, lógicas, desplazamientos) que en ella se llevan a cabo.
2. Vienen de la decodificación de la instrucción llevada a cabo en la unidad de control.
3. Leer, decodificar y hacer que se ejecuten las instrucciones de memoria; controlando y dirigiendo para ello el funcionamiento de la ruta de datos.
4. En la U.C. cableada la obtención de las señales de control a partir de la instrucción se realiza en un circuito decodificador, diseñado e implementado para unas instrucciones y una palabra de control dadas. En la U. C. microprogramada sin embargo, parte de la instrucción se utiliza para direccionar una memoria en la que se almacenan las correspondientes palabras de control.
5. Porque una operación de lectura o escritura en memoria requeriría dos accesos a memoria, lo que resultaría inviable en un único ciclo de memoria.
6. Al ser de ciclo múltiple, la unidad de control requiere ciertos registros adicionales (registros no accesibles al usuario en la unidad de control por ejemplo), lo que implica señales de control adicionales, que deberán aparecer en la palabra de control.
7. Se añaden ceros a la izquierda del operando. Cuando se trata de instrucciones de "operando inmediato". Dicho operando (contenido en la instrucción) tiene menos bits que los operandos de los registros, pero para poder realizar operaciones entre ambos es necesario que tengan el número de bits. El bloque extensión de signo realiza algo parecido, alarga el número de bits del offset para modificar el contador de programa y realizar bifurcaciones; la diferencia es que en este caso al poder ser el offset un número positivo o negativo, es necesario mantener el signo, por lo que se añaden ceros o unos según sea el MSB.
8. En principio no, ya que podríamos tener  $2^5$  diferentes instrucciones. En el caso de ciclo múltiple la ejecución de cada instrucción requerirá múltiples microinstrucciones y por ende, múltiples palabras de control (y cada una de ellas necesitará una posición de memoria).
9. Para la U.C. cableada será necesario modificar físicamente (tal vez rediseñándolo por completo) el decodificador; en el caso de la microprogramada sólo hay que reprogramar la memoria.

## Ejercicios Tema 2

---

### Diagrama de flujo:

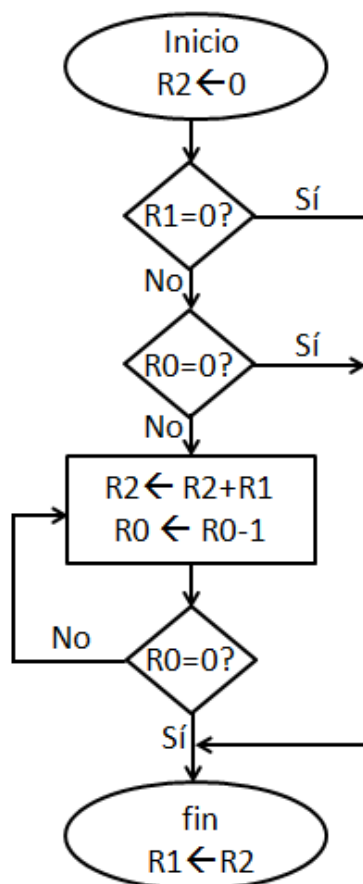
1. Supongamos un juego de instrucciones que no incluye una instrucción de multiplicación, pero sí una de suma. Realiza el diagrama de flujo del algoritmo que realiza la multiplicación en base a la suma. Los números ( $\geq 0$ ) están en los registros R1 y R0 (podemos suponer que ya están ahí, o que se introducen por los puertos 1 y 2), y el resultado ha de quedar en el registro que contiene el primer multiplicando.

MUL:  $R1 \leftarrow R1 \times R0$

2. Dibuja el diagrama de flujo del algoritmo que calcula el factorial de un número (que está en un registro o que se introduce por un puerto).

3. Piensa en los registros que vas a utilizar en cada caso para combinando ambos algoritmos implementar el cálculo del factorial en el 8085.

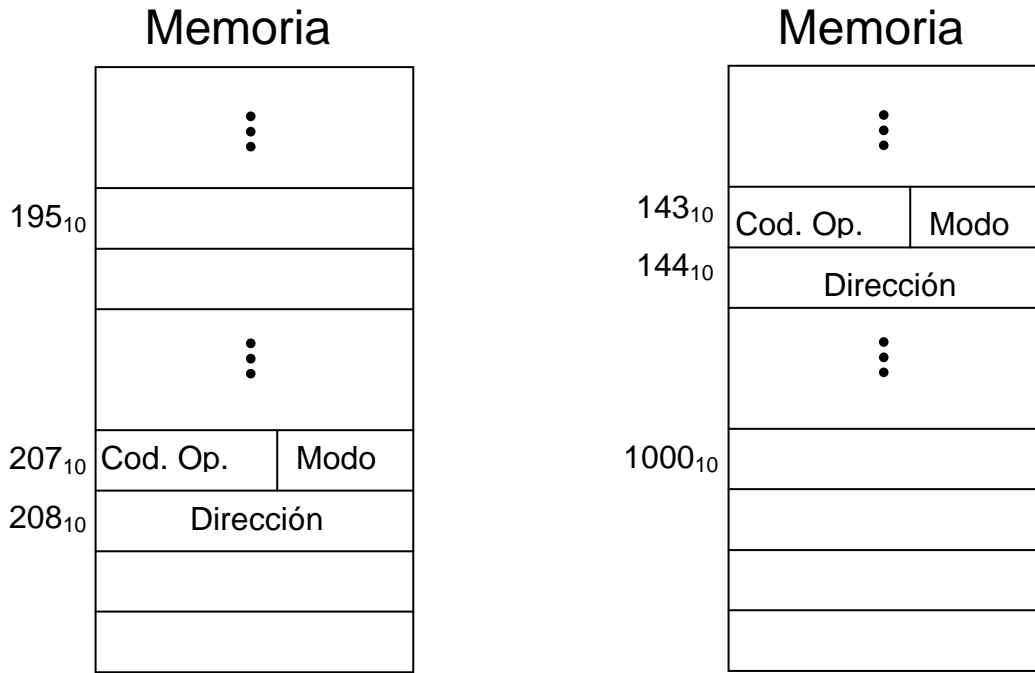
### Diagrama de flujo (solución ejercicio 1)





## Juego de instrucciones y direccionamiento

- 1- Si tenemos instrucciones de 32 bits con los siguientes campos: *Código de Operación* (6 bits), *R1* (6 bits), *R2* (6 bits) y *Operando Inmediato* (14 bits)...
  - a) ... ¿cuál es el número máximo de operaciones?
  - b) ... ¿cuántos registros pueden direccionarse?
  - c) ... si los operandos son números binarios sin signo, ¿entre qué valores estarán dichos operandos?
  - d) ... si el 1<sup>er</sup> bit del campo *Operando* (el 13<sup>er</sup> bit en total) representa el signo, ¿entre qué valores estarán los operandos para las representaciones *Signo y Magnitud* y *Complemento a 2*?
  
- 2- Memoria de 32 bits, bloque de registros de 32 registros, juego de instrucciones formado por 110 instrucciones diferentes (formato de instrucción único):
  - a) ¿Cuántos bits se necesitan para el campo *Código de Operación*?
  - b) ¿Cuántos bits quedan para el operando?
  - c) Si se utiliza como dirección de memoria sin signo, ¿cuántas posiciones de memoria pueden direccionarse?
  - d) Si se utiliza como operando inmediato (representado en complemento a 2), ¿cuáles serán el número menor y mayor representables?
  
- 3- Tenemos instrucciones de 32 bits, pero la longitud del campo Código de Operación no es siempre la misma: si el código de operación empieza por 0, la longitud es de cuatro bit; si empieza por 10 es de 6 y de 8 si empieza por 11. ¿Cuántos códigos de operación diferentes tenemos?
  
- 4- En memoria tenemos una instrucción de 2 palabras que indica que hay que realizar una operación con un dato de memoria dejando el resultado en el mismo lugar. Si se utiliza direccionamiento indirecto, ¿cuántos accesos a memoria serán necesarios?
  
- 5- La instrucción almacenada en las posiciones 207, 208 de la memoria (figura de la izquierda) utiliza un operando contenido en la dirección 195. Si el campo **modo** de la instrucción nos indica que el direccionamiento es **relativo**, para el campo **dirección** ¿cuál será:
  - a) el valor decimal?
  - b) el valor binario (16 bit y Ca2)?
  
- 6- Lo mismo pero ahora la instrucción está en las posiciones 143, 144 (figura de la derecha) y hay que saltar a la posición 1000.



7- ¿Cuál será la dirección efectiva para los siguientes modos de direccionamiento?

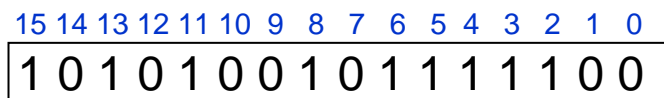
⋮
0100   CO   modo
0101   211
⋮

R1=189

Direccionamiento	Directo	Inmediato	Relativo	Reg. Indir.	Indexado (R1)
Dir. efectiva					

8- Siendo A=00110101 y B=10111001, realiza la AND, OR y XOR de A y B

9- Dado el siguiente valor de 16 bit, ¿qué operaciones hay que realizar y con qué valores para sin modificar el resto de bits...



- a)... poner los 8 bits de menor peso a '1'?
- b)... negar los bits de las posiciones impares (peso 0, 2, 4...)?
- c)...poner a '0' los bits de las posiciones impares

## Juego de instrucciones y direccionamiento (solución)

1. a) 64 operaciones      b) 64 registros      c)  $0 \leq N \leq 2^{14}-1$   
d)  $(-2^{13}+1) \leq N \leq 2^{13}-1$  (S&M) y  $(-2^{13}) \leq N \leq 2^{13}-1$  (Ca2)
2. a) 7 bits (Cod. Op.)      b)  $32(\text{total}) - 7(\text{Cod. Op.}) - 5(\text{reg.}) = 20$   
c)  $2^{20}$       d)  $\text{mín} = (-2^{19})$ ;  $\text{máx} = (-2^{19}-1)$
3.  $2^3+2^4+2^6 = 88$  códigos de operación
4. Leer instrucción (2) + obtener operando (2) + escribir resultado (1) = 5
5. Suponiendo que el contador de programa ya tiene la dirección de la siguiente instrucción (PC=209)  $\rightarrow 209 + \text{dir.} = 195 \rightarrow$   
a)  $\text{dir.} = -14_{10}$       b)  $1111111111110010_2$
6. PC=145  $\rightarrow$  a)  $\text{dir.} = 855_{10}$       b)  $0000001101010111_2$
7. Suponiendo direcciones en decimal y PC=102  

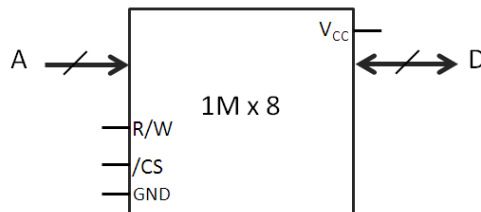
Dir. Efec.	211	101	$211+102=313$	189	$189+102=291$
------------	-----	-----	---------------	-----	---------------
8. a) 00110001 (AND)      b) 10111101 (OR)      c) 10001100 (XOR)
9. Con las siguientes operaciones y máscaras se realizan las operaciones requeridas sin modificar el resto de bits y para cualquier dato (no sólo para el del ejemplo)  
a) Operación OR con 0000000011111111  
b) Operación XOR con 0101010101010101  
c) Operación AND con 1010101010101010

## Ejercicios Tema 4

---

### Memoria

- Una memoria de semiconductor viene especificada como 2Kx8.
  - ¿Cuántas palabras almacena?
  - ¿Cuál es el tamaño de la palabra?
  - ¿Cuántos bits almacena la memoria?
- Sea una memoria de 12M bits, a la que se accede en palabras de 16 bits.  
¿Cuántas líneas de dirección son necesarias?
- Sea una memoria de 4.194.304 bits, a la que se accede en palabras de 8 bits (1 byte). ¿Cuántas líneas de dirección son necesarias?
- Un circuito integrado de memoria tiene una capacidad de 32Kx16. Indicar:
  - ¿Cuántas líneas de dirección tiene?
  - ¿Cuál será su capacidad en bytes? ¿Y en bits?
  - ¿Cuántas líneas de datos tiene?
- Cuántas líneas de dirección tendrá una memoria, con la misma capacidad que la del ejercicio anterior, en la que se direcciona a byte.
- ¿Cuántos terminales requiere el circuito integrado de una memoria como la de la figura, si emplea líneas de E/S de datos comunes?



- Para los siguientes supuestos 1)-5)
  - Queremos implementar una memoria de 2 MB con chips de 512 KB
  - 2 MB con chips de 512 K x 4
  - 2 MB con chips de 256 KB
  - 2Mx16 con chips de 512 KB
  - 2Mx16 con chips de 512 K x 4

Responde las preguntas a)-f)

- ¿Cuántos bits se necesitan para direccionar la memoria que se quiere implementar? (Anchura del bus de direcciones final)
- ¿Cuántos bits se necesitan para direccionar un chip de memoria?
- Anchura del bus de datos final
- Anchura del bus de datos de cada chip

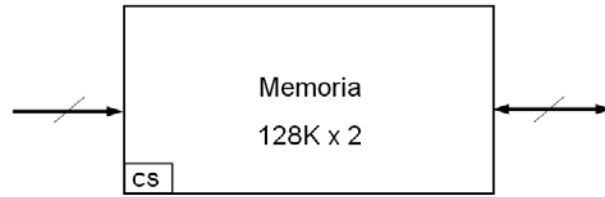
- e) ¿Cuántos chips se necesitan (para lograr ese tamaño de memoria x para lograr esa anchura de palabra) en total?
- f) ¿Cuántos bits son necesarios para direccionar esos chips (para los "chip select")? De todos los bits disponibles en el bus de direcciones (de la memoria final) ¿cuáles utilizarás para seleccionar el(los) chip(s) necesario(s)?
- g) Realiza las conexiones

RESPUESTAS:

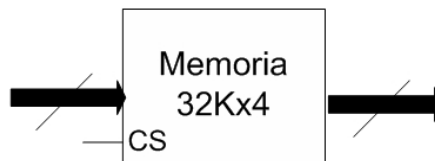
	a	b	c	d	e	f
1						
2						
3						
4						
5						

- 8. Un circuito integrado (CI) de memoria RAM estática tiene una capacidad de 16Kx2, salida y entrada de datos separadas, una entrada de selección de circuito activa a nivel bajo (/CS) y una entrada R/W. Indicar cómo combinar varios de dichos circuitos para formar una memoria de 16Kx8.
- 9. Se desea construir una memoria ROM de 32Kx8 empleando circuitos integrados PROM de 8Kx8. Indicar el número de circuitos PROM necesarios y cómo combinarlos para obtener la capacidad total deseada.
- 10. La MC de un sistema tiene un tiempo de acceso de 5 ns, siendo el tiempo de acceso la MP de 50 ns. El 30% de las peticiones son para escritura (resto lectura). Si la razón de aciertos de la MC es de 0,8 y se emplea una política de actualización de escritura inmediata ("write through"), determinar:
  - a. El tiempo medio de acceso atendiendo solo a ciclos de lectura
  - b. El tiempo medio de acceso teniendo en cuenta también los ciclos de escritura
- 11. Diseñar un sistema paginado de memoria virtual de acuerdo con las siguientes especificaciones:
  - Tamaño del espacio virtual direccionable = 64K
  - Tamaño del espacio de direcciones físicas = 8K
  - Tamaño de la página = 512 palabras
  - Longitud total de los registros de la tabla de páginas = 24 bits

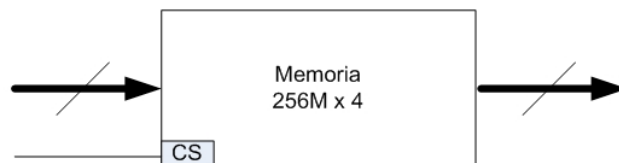
12. Queremos diseñar una memoria de 256 kB utilizando módulos como el de la figura. ¿Cuántos dispositivos de memoria necesitamos? Realiza las conexiones.



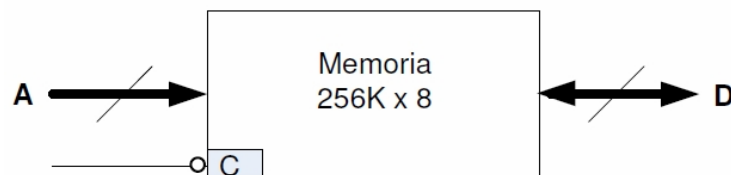
13. Queremos diseñar una memoria principal de 128 KB utilizando módulos como el de la figura. ¿Cuántos dispositivos de memoria necesitamos? Realizar las conexiones.



14. Queremos diseñar una memoria principal de 1 GB utilizando módulos como el de la figura. ¿Cuántos dispositivos de memoria necesitamos? Realizar las conexiones



15. Se pretende diseñar una memoria de 1MB para lo que se dispone de chips de memoria como el de la figura. Dibujar la implementación de la memoria explicando las decisiones (por qué ese número de circuitos integrados, por qué esa conexión...)



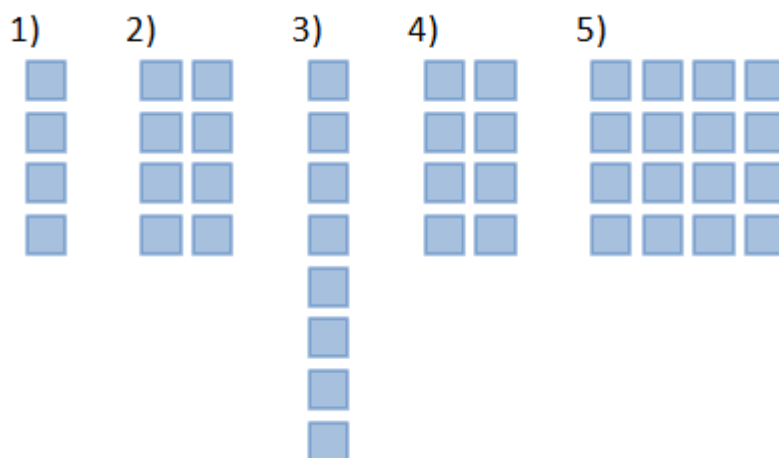
16. Memoria caché (teoría): una palabra es  $K$  veces referenciada en un corto periodo de tiempo; la primera vez no está en caché y ha de traerse de memoria principal. Si el tiempo de acceso de la caché es  $C$  y el tiempo de acceso de la memoria principal es  $M$ , definir la tasa de acierto y el tiempo medio de acceso.

## Ejercicios Tema 4: Memorias (soluciones)

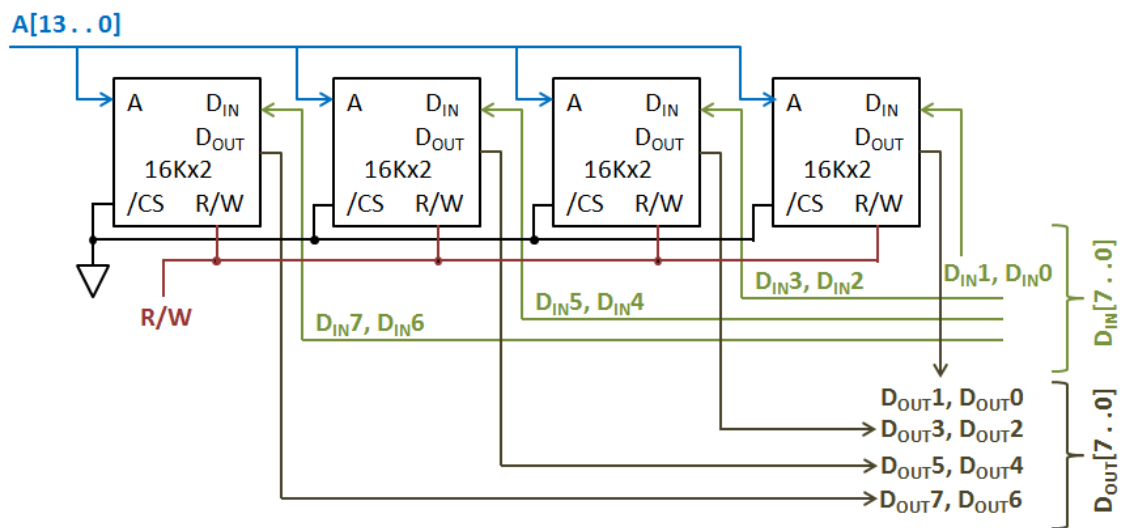
- 1) a) 2048 palabras    b) 8 bits (byte)    c) 16384 bits (2048x8)
- 2) 12 Mbits= 12582912 bits → 786432 posiciones → son necesarias 20 líneas de dirección (aunque con 20 líneas pueden direccionarse 1048576 posiciones, más de las que tenemos, con 19 no es suficiente, pues sólo se direccionan 524288)
- 3)  $4194304/8=524288$  posiciones → se necesitan 19 líneas de dirección
- 4) a) 15 líneas de dirección  
b) 65536 bytes = 524288 bits  
c) 16 líneas de datos
- 5) 64Kx8 → 16 líneas de dirección
- 6) 1Mx8 → 20 para direcciones, 8 para datos, 1 R/W, 1 /CS y 2 alimentación → 32 terminales

### 7) Tabla soluciones

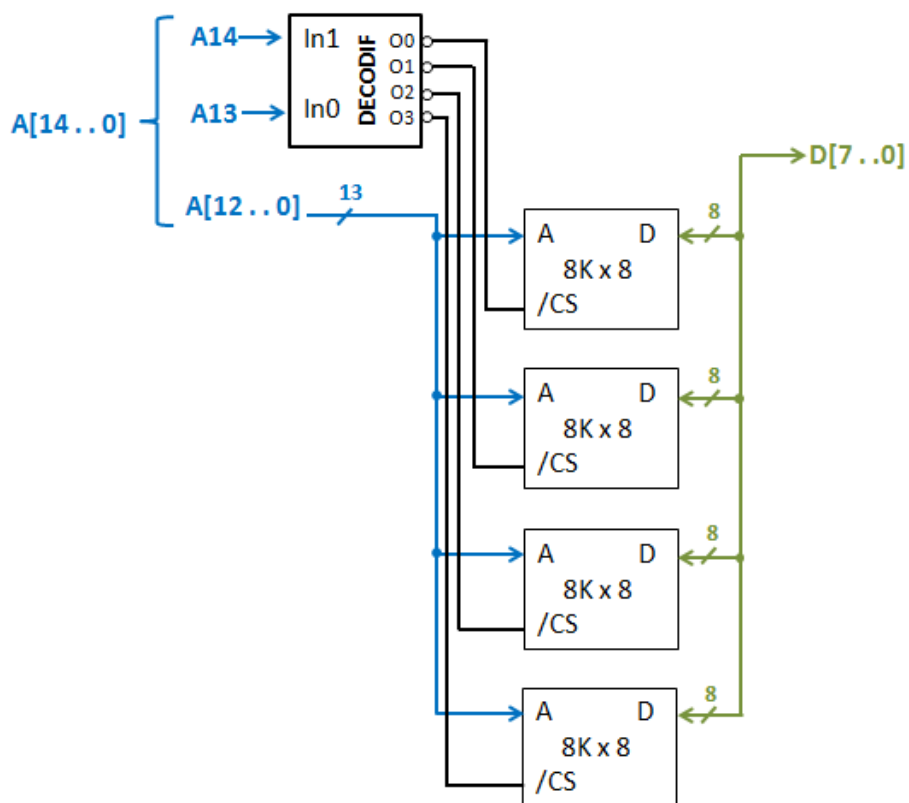
	a (dir. final)	b dir. chip)	c (datos final)	d (datos chip)	e (nº chips)	f (bits para CS)
1	21	19	8	8	4	2 bits (A20, A19)
2	21	19	8	4	8	2 bits (A20, A19)
3	21	18	8	8	8	3 bits (A21, A20, A19)
4	21	19	16	8	8	2 bits (A20, A19)
5	21	19	16	4	16	2 bits (A20, A19)



8) Ampliación tamaño de palabra



9) Ampliación posiciones de memoria

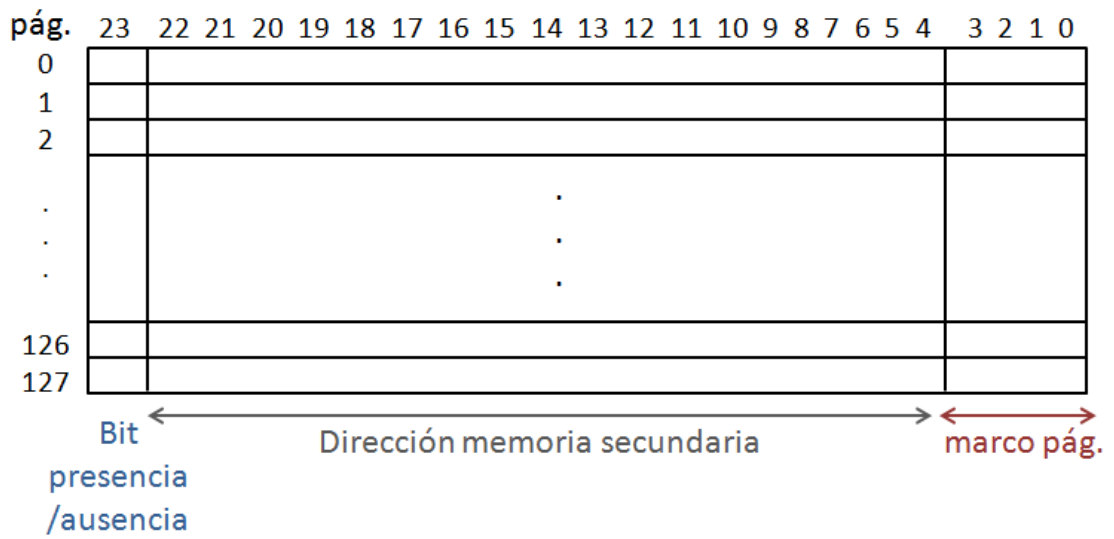


10)  $t_{MC}=5\text{ns}$ ,  $t_{MP}=50\text{ ns}$ , 30% escritura, 70% lectura, 80% hit, 20% miss

- a)  $t_{lectura}=0,8 \cdot t_{MC}+0,2 \cdot (t_{MC}+t_{MP})=15\text{ ns}$
- b)  $t=0,3 \cdot t_{lectura} + 0,7 \cdot t_{escritura}=25,5\text{ ns}$



- 11) Si el espacio virtual direccionable es de 64K y la memoria física de 8K; para un tamaño de página de 512 palabras tendremos 128 páginas y 16 marcos de página.



- Tabla de páginas -

## Ejercicios Tema 6

---

### Buses

1. Si un protocolo de bus tiene las siguientes señales: CLK, A[15..8], AD[7..0] Y R/W
  - a) Características del protocolo de comunicación que puedan deducirse
  - b) Se ha de sustituir la memoria por una más lenta ¿qué modificaciones harías?

2. Marca con una x las señales características de cada tipo de sincronización:

Señales	Síncrono	Asíncrono	Semisíncrono
Clk			
Wait, Busy			
IRDY, Msyn			
TRDY, Ssyn			

3. Si un protocolo de bus tiene las siguientes señales: CLK, A[7..0], D[7..0], R/W y READY:
  - a. Características del protocolo de comunicación que puedan deducirse
  - b. Si una operación de escritura requiere 3 ciclos de reloj (en el mejor de los casos), siendo la frecuencia del mismo de 8 MHz, calcula el tiempo de escritura (en el mejor de los casos)
  - c. ¿Cuál es el espacio de direccionamiento? ¿Qué cambios harías para que el espacio de direccionamiento pase a ser de 64 kB con el mínimo número de líneas necesario?
4. Un bus síncrono que trabaja con un reloj de 8 MHz tarda 3 ciclos para realizar una lectura desde cualquier dispositivo. Para mejorar el rendimiento se cambia a un reloj de 9 MHz, lo que implica que algunos dispositivos requieran 4 ciclos en lugar de 3, volviéndose semisíncrono. Si de media se tardan 3,5 ciclos ¿merece la pena el cambio? Justifica la respuesta.
5. Un bus síncrono que trabaja a una frecuencia de reloj de 50 MHz tarda 3 ciclos para hacer una lectura de una palabra de 16 bit, y 4 ciclos para hacer una escritura.
  - a. Ancho de banda del bus para lectura. Ídem para escritura.
  - b. ¿Cuánto tiempo necesitará para leer 1 MB? ¿Y para escribirlo?

## Ejercicios Tema 6. Buses (soluciones)

1.

a) Síncrono (CLK) y multiplexado (las mismas líneas a veces llevan direcciones a veces datos, AD[7..0])

b) Añadir una línea que lo haga semisíncrono, metiendo un ciclo de espera solo cuando sea necesario.

2.

Señales	Síncrono	Asíncrono	Semisíncrono
Clk	X		X
Wait, Busy			X
IRDY, Msyn		X	
TRDY, Ssyn		X	

3.

a) Semisíncrono (CLK y READY) y buses dedicados (líneas separadas para direcciones y datos.

b)  $f_{CLK} = 8 \text{ MHz} \rightarrow T_{CLK} = 125 \text{ ns} \rightarrow 3 \times T_{CLK} = 375 \text{ ns}$

c) Bus direcciones 8 bits  $\rightarrow$  pueden direccionarse 256 posiciones. Para 64 K posiciones se necesitan 16 bits. Una opción sería doblar el bus de direcciones (de 8 a 16 bits), pero la opción que requiere menos líneas consiste en multiplexar el bus de direcciones, de modo que a sus 8 líneas veces lleven direcciones y a veces datos. Esto sólo requiere añadir una línea de control que indique en qué estado está el bus multiplexado.

4. Si  $f_{CLK} = 8 \text{ MHz} \rightarrow T_{CLK} = 125 \text{ ns} \rightarrow t_{lectura} = 3 \times T_{CLK} = 375 \text{ ns}$

Si  $f_{CLK} = 9 \text{ MHz} \rightarrow T_{CLK} = 111,1 \text{ ns} \rightarrow t_{medio} = 3,5 \times T_{CLK} = 388,9 \text{ ns} > 375 \text{ ns}$  por lo que no merece la pena.

5. Si  $f_{CLK} = 50 \text{ MHz} \rightarrow T_{CLK} = 20 \text{ ns} \rightarrow t_{lectura} = 60 \text{ ns}$  y  $t_{escritura} = 80 \text{ ns}$

a) Ancho de banda de lectura:  $0,267 \cdot 10^9 \text{ bps} \rightarrow 254,32 \text{ Mbps}$   
 Ancho de banda de escritura:  $0,2 \cdot 10^9 \text{ bps} \rightarrow 190,73 \text{ Mbps}$

b) Tiempo para leer 1MB=31,46 ms  
 Tiempo para escribir 1MB=41,95 ms