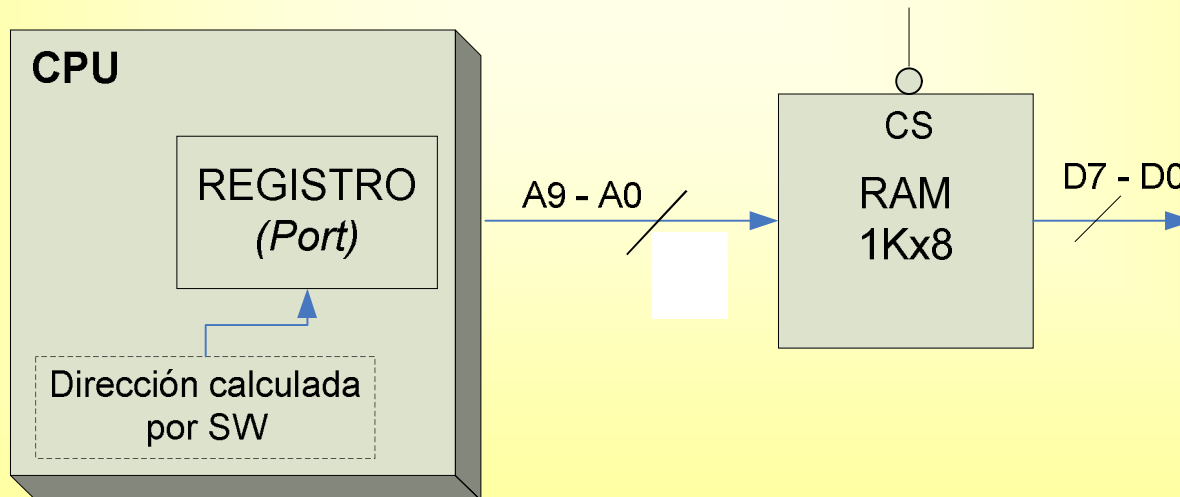
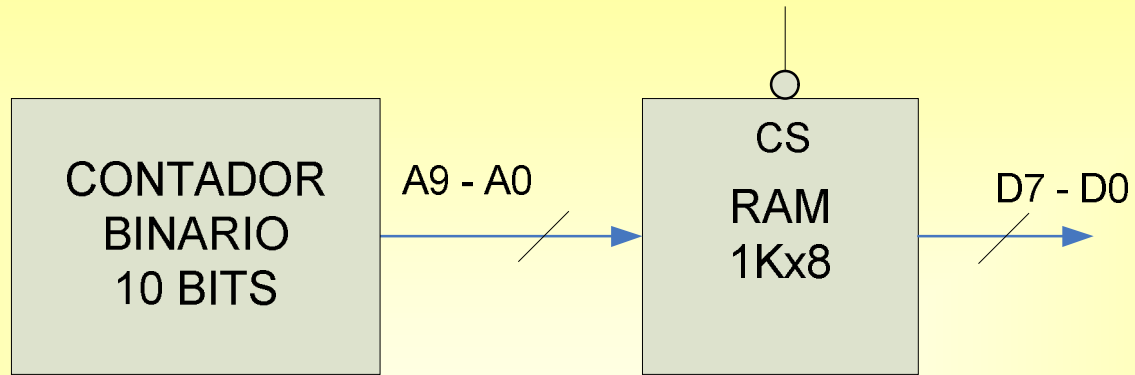


Departamento de Electrónica
Electrónica Digital

Mapas de memoria

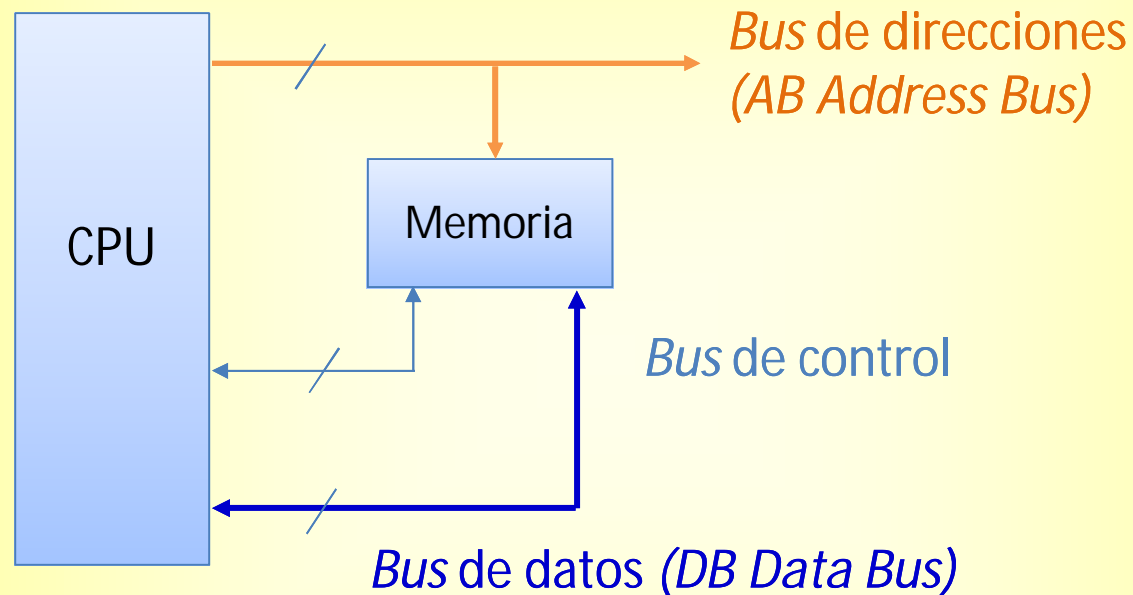
Bioingeniería
Facultad de Ingeniería - UNER

Direccinamiento de las memorias



Espacio de direcciones

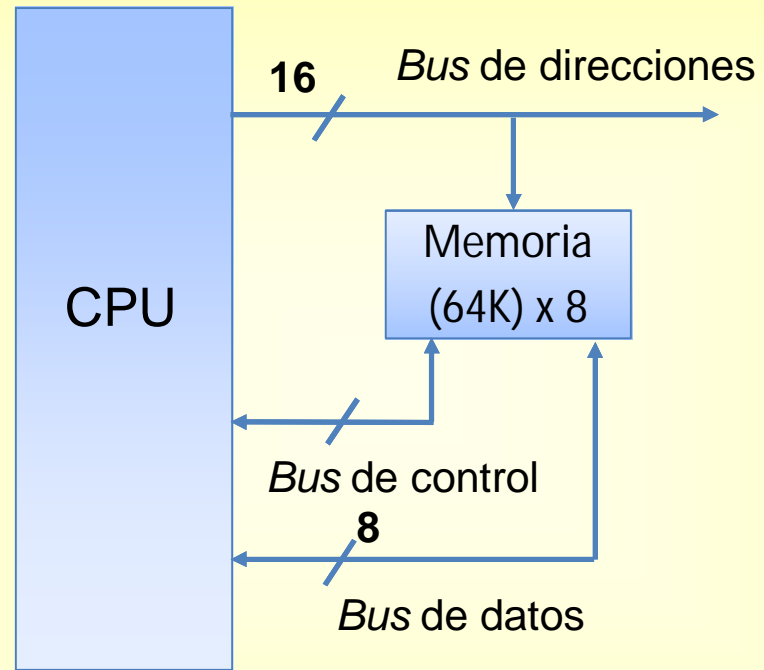
Es la **capacidad de direccionamiento** de la CPU y está definido por el tamaño del *bus* de direcciones.

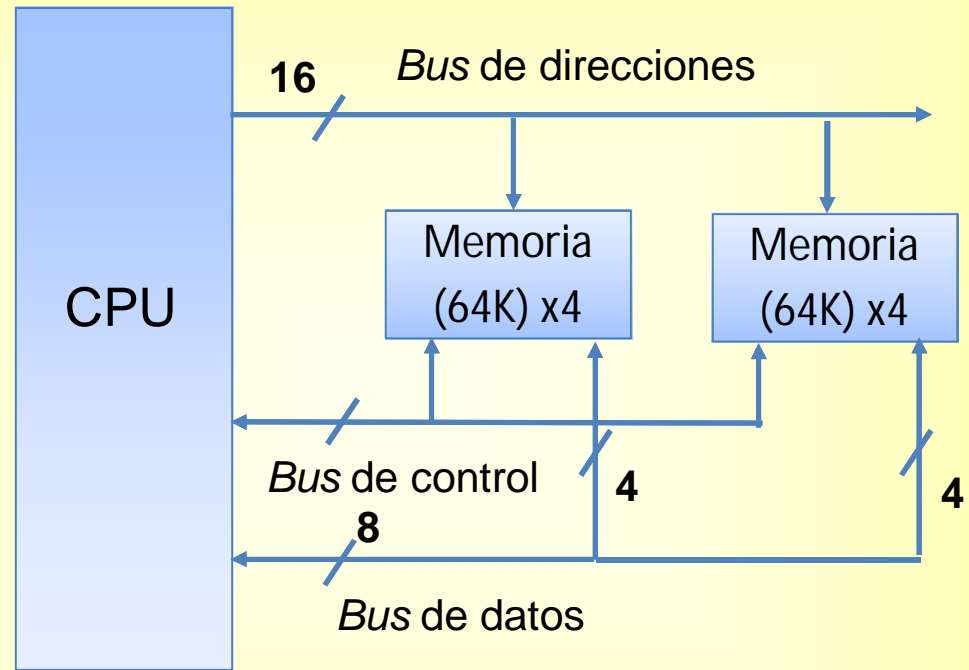


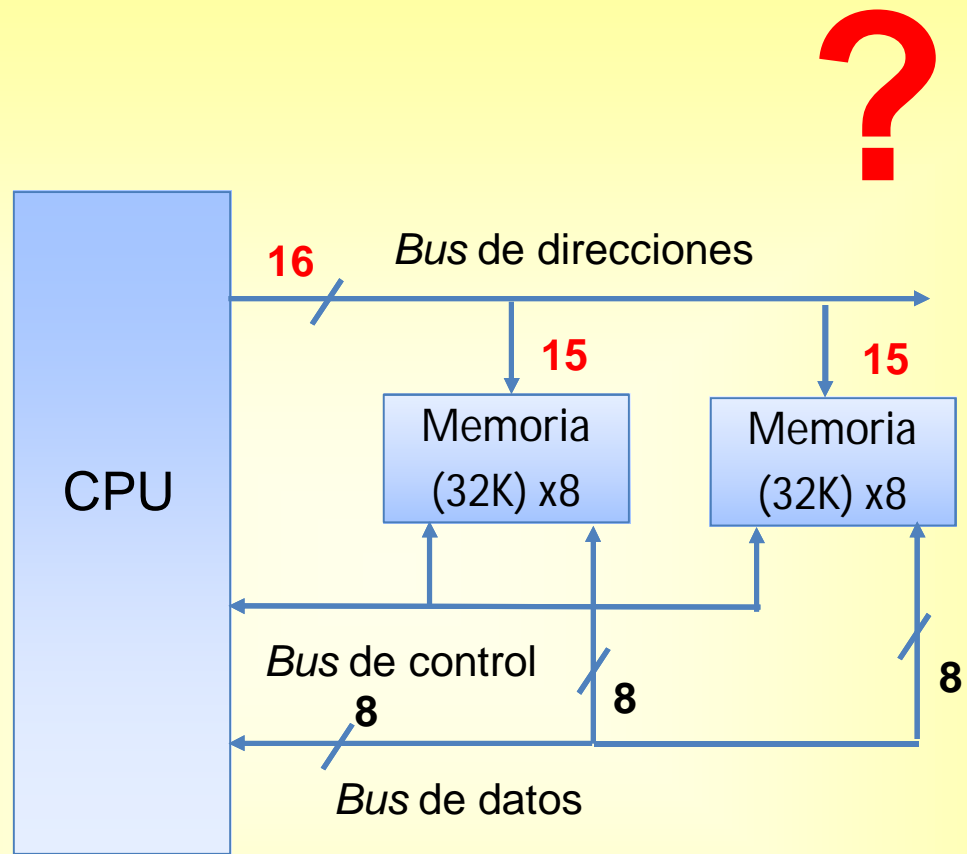
Ejemplo: AB de 16 líneas (A_{15}, \dots, A_0)

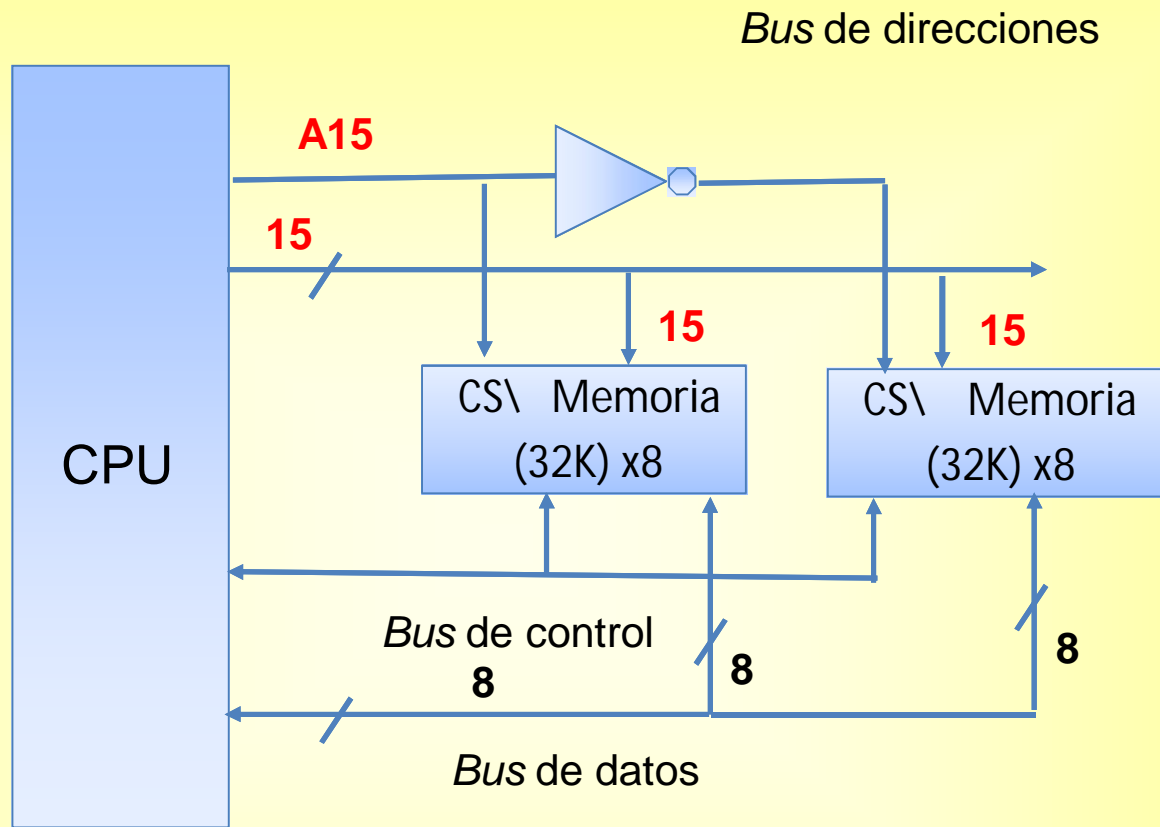
\Rightarrow H0000 a HFFFF

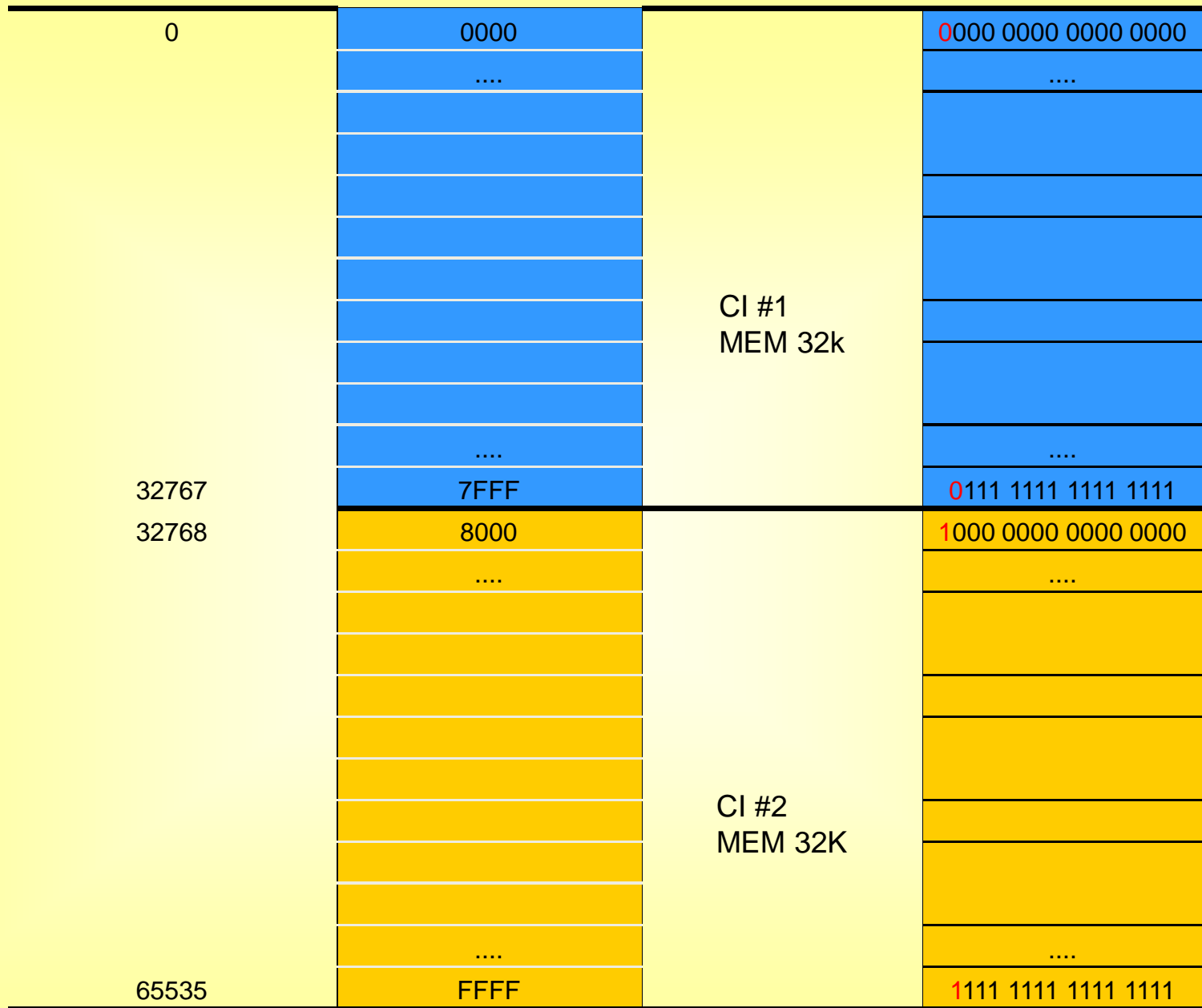
$\Rightarrow 0_{10}$ a $65535_{10} = 64K$ palabras





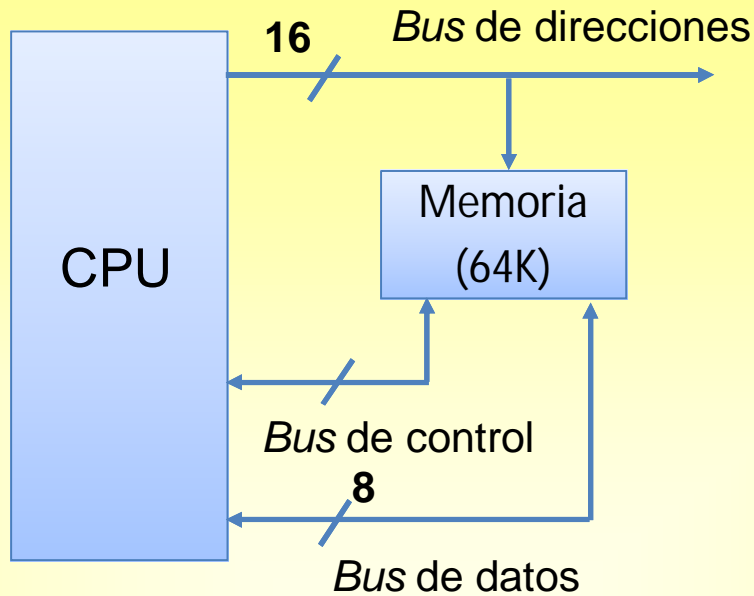




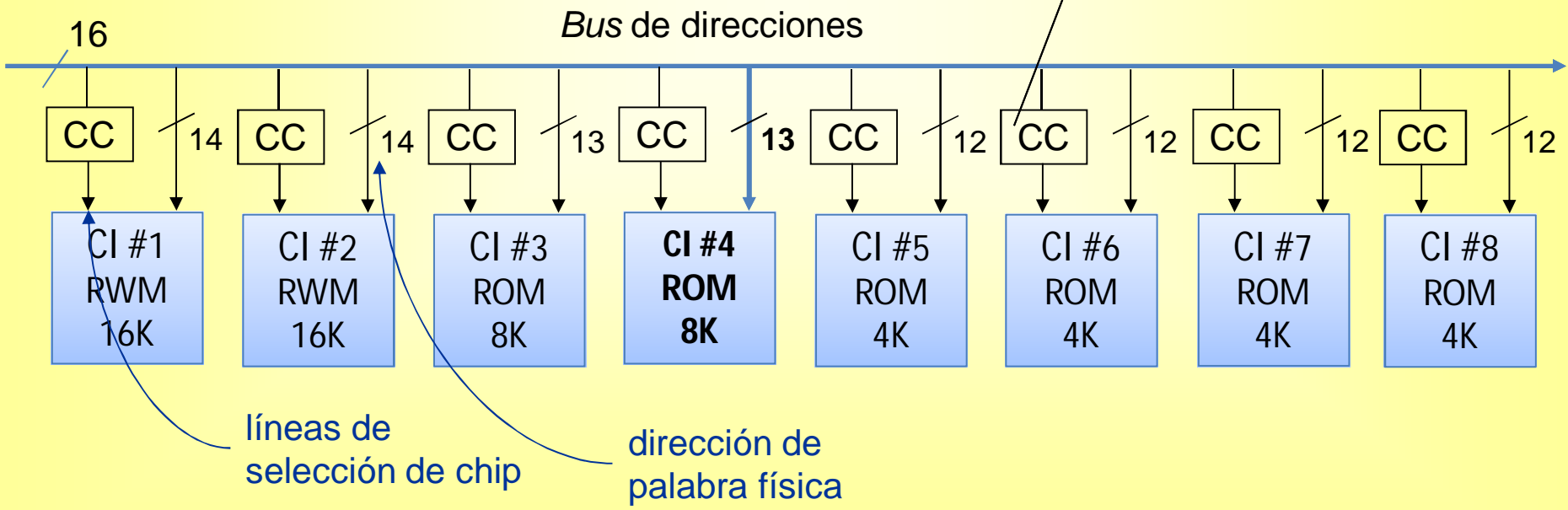


Referencias

- H03FF = 1K
- H07FF = 2K
- H0FFF = 4K
- H1FFF = 8K
- H3FFF = 16K
- H7FFF = 32K



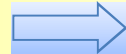
Circuito decodificador de selección de chip



Mapa de memoria

Es la **distribución** de cada chip de memoria dentro del espacio de direcciones y está determinado por el **rango de direcciones con el cual se selecciona cada chip** mediante el CC

Dirección
HA018



0	0000	CI #1 RWM 16K
	
16383	3FFF	
16384	4000	CI #2 RWM 16K
	
32767	7FFF	
32768	8000	CI #3 ROM 8K
	
40959	9FFF	
40960	A000	CI #4 ROM 8K (13 bits)
	
49151	BFFF	
41152	C000	CI #5 ROM 4K
	
53247	CFFF	
53248	D000	CI #6 ROM 4K
	
57343	DFFF	
57344	E000	CI #7 ROM 4K
	
61439	EFFF	
61440	F000	CI #8 ROM 4K
	
65535	FFFF	

Dirección del
mapa de
memoria

Chip 4
ROM
8K

Dirección
interna del CI

HA000

H0000 0

....

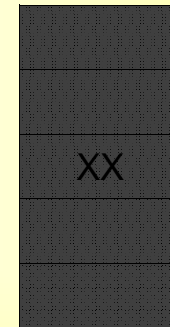
HA018

H0018 24

....

HBFFF

H1FFF 8191



1010 0000 0001 1000

0000 0000 0001 1000

Referencias

- H03FF = 1K
- H07FF = 2K
- H0FFF = 4K
- H1FFF = 8K
- H3FFF = 16K

5/7/5/2019

0	0000	CI #1 RWM 16K	0000 0000 0000 0000

16383	3FFF		0011 1111 1111 1111
16384	4000	CI #2 RWM 16K	0100 0000 0000 0000

32767	7FFF		0111 1111 1111 1111
32768	8000	CI #3 ROM 8K	1000 0000 0000 0000

40959	9FFF		1001 1111 1111 1111
40960	A000	CI #4 ROM 8K (13 bits)	1010 0000 0000 0000

49151	BFFF		1011 1111 1111 1111
41152	C000	CI #5 ROM 4K	1100 0000 0000 0000

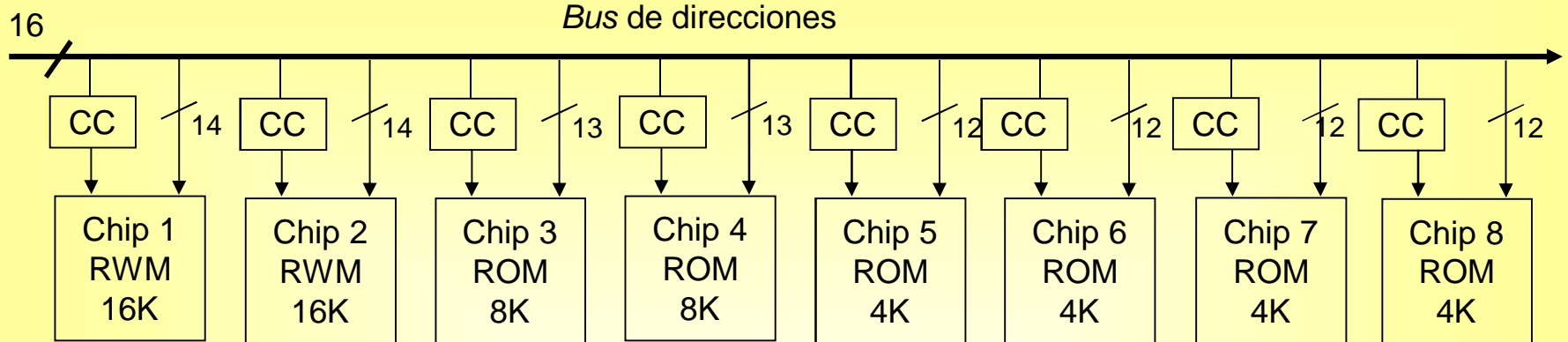
53247	CFFF		1100 1111 1111 1111
53248	D000	CI #6 ROM 4K	1101 0000 0000 0000

57343	DFFF		1101 1111 1111 1111
57344	E000	CI #7 ROM 4K	1110 0000 0000 0000

61439	FFFF		1110 1111 1111 1111
61440	F000	CI #8 ROM 4K	1111 0000 0000 0000

65535	FFFF		1111 1111 1111 1111

Selección de cada CI dentro del mapa



La **parte alta** del AD se usa para **seleccionar** cada memoria por medio de un circuito decodificador (combinacional); la **parte baja** se usa para **direccionar**.

A_{15}	A_{14}	A_{13}	A_{12}	Chip	Bits para direccionar
0	0	-	-	1 (16K)	14 bits $A_{13} \dots A_0$
0	1	-	-	2 (16K)	
1	0	0	-	3 (8K)	13 bits $A_{12} \dots A_0$
		1	-	4 (8K)	
1	1	0	0	5 (4K)	12 bits $A_{11} \dots A_0$
		0	1	6 (4K)	
		1	0	7 (4K)	
		1	1	8 (4K)	

$$CS_{1/} = A_{15} + A_{14}$$

$$CS_{2/} = A_{15} + A_{14}'$$

$$CS_{3/} = A_{15}' + A_{14} + A_{13}$$

$$CS_{4/} = A_{15}' + A_{14} + A_{13}'$$

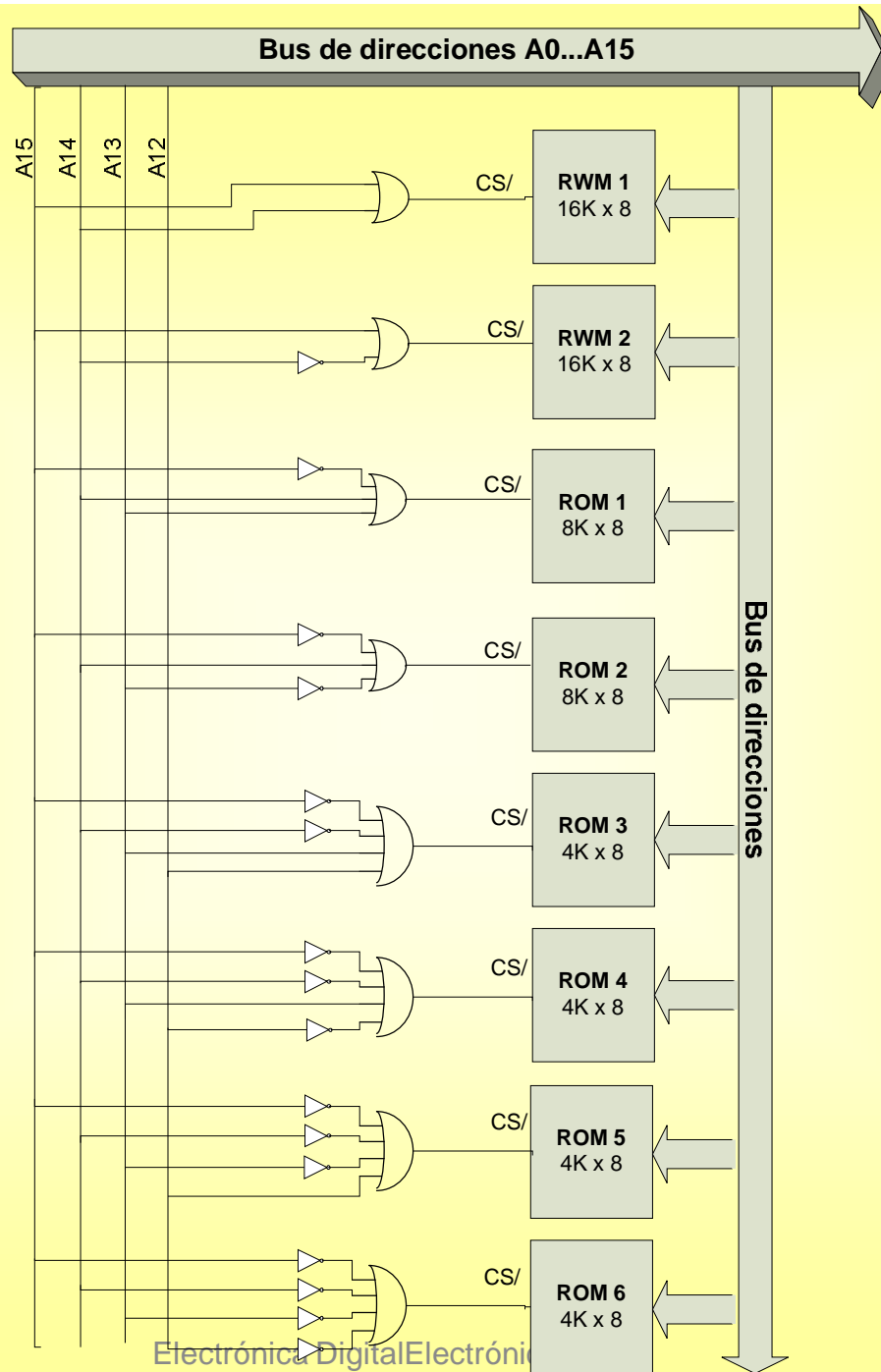
$$CS_{5/} = A_{15}' + A_{14}' + A_{13} + A_{12}$$

$$CS_{6/} = A_{15}' + A_{14}' + A_{13} + A_{12}'$$

$$CS_{7/} = A_{15}' + A_{14}' + A_{13}' + A_{12}$$

$$CS_{8/} = A_{15}' + A_{14}' + A_{13}' + A_{12}'$$

Direcciones mapa	Bits de selección del AD				Salidas del CC (decodificador)								Chip		
	A ₁₅	A ₁₄	A ₁₃	A ₁₂	CS ₁	CS ₂	CS ₃	CS ₄	CS ₅	CS ₆	CS ₇	CS ₈			
0000 3FFF	0	0	-	-	0	1	1	1	1	1	1	1	1 (16K) 14 bits		
4000 7FFF	0	1	-	-	1	0	1	1	1	1	1	1	2 (16K)		
8000 9FFF	1	0	0	-	1	1	0	1	1	1	1	1	3 (8K) 13 bits		
A000 BFFF			1	-	1	1	1	0	1	1	1	1	1	4 (8K)	
C000 CFFF	1	1	0	0	1	1	1	1	0	1	1	1	5 (4K) 12 bits		
D000 DFFF			0	1	1	1	1	1	1	0	1	1	1	6 (4K)	
E000 EFFF			1	0	1	1	1	1	1	1	1	0	1	1	7 (4K)
F000 FFFF			1	1	1	1	1	1	1	1	1	1	1	0	1



Métodos de selección de las memorias

Selección decodificada

Sistemas grandes

- Se basa en decodificar las líneas de dirección para la selección
- Reduce la cantidad de líneas de dirección del banco final
- Permite el aprovechamiento integral del bus de direcciones

Selección lineal

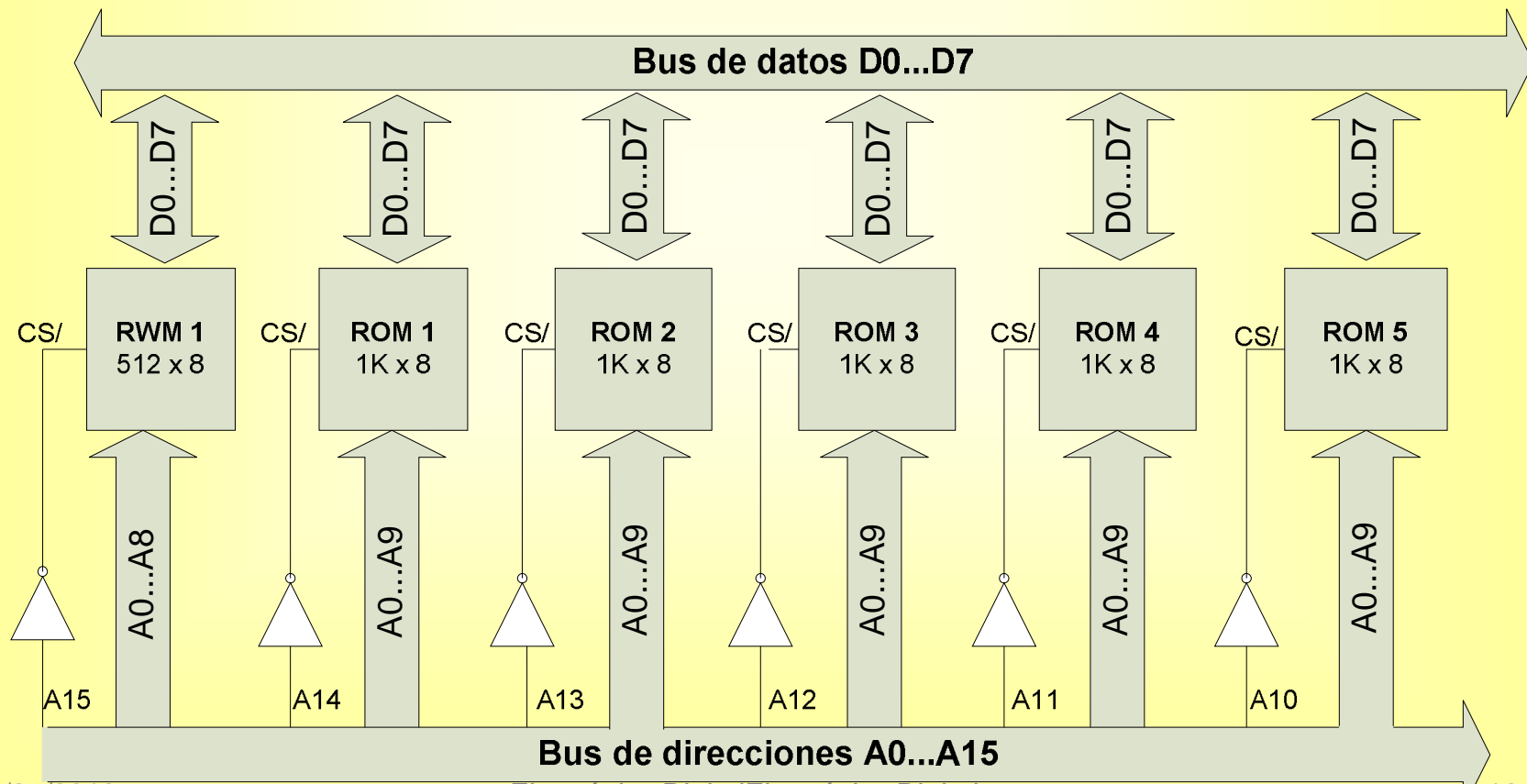
Sistemas pequeños, cuando sobran líneas del bus de direcciones

- Se necesitan tantas líneas de selección como chips de memoria contenga el banco

Ejemplo de selección lineal

Se requiere un banco de memoria de 5 ROMs de 1 K x 8 y 1 RWM de 512 x 8.

- Para direccionar la RWM se requieren 9 bits
- Para direccionar las ROM se requieren 10 bits
- Para seleccionar (método lineal) se requieren 6 bits (1 por cada CI)
- Se requiere un total de 16 líneas de dirección (AB)



Mapa de memoria

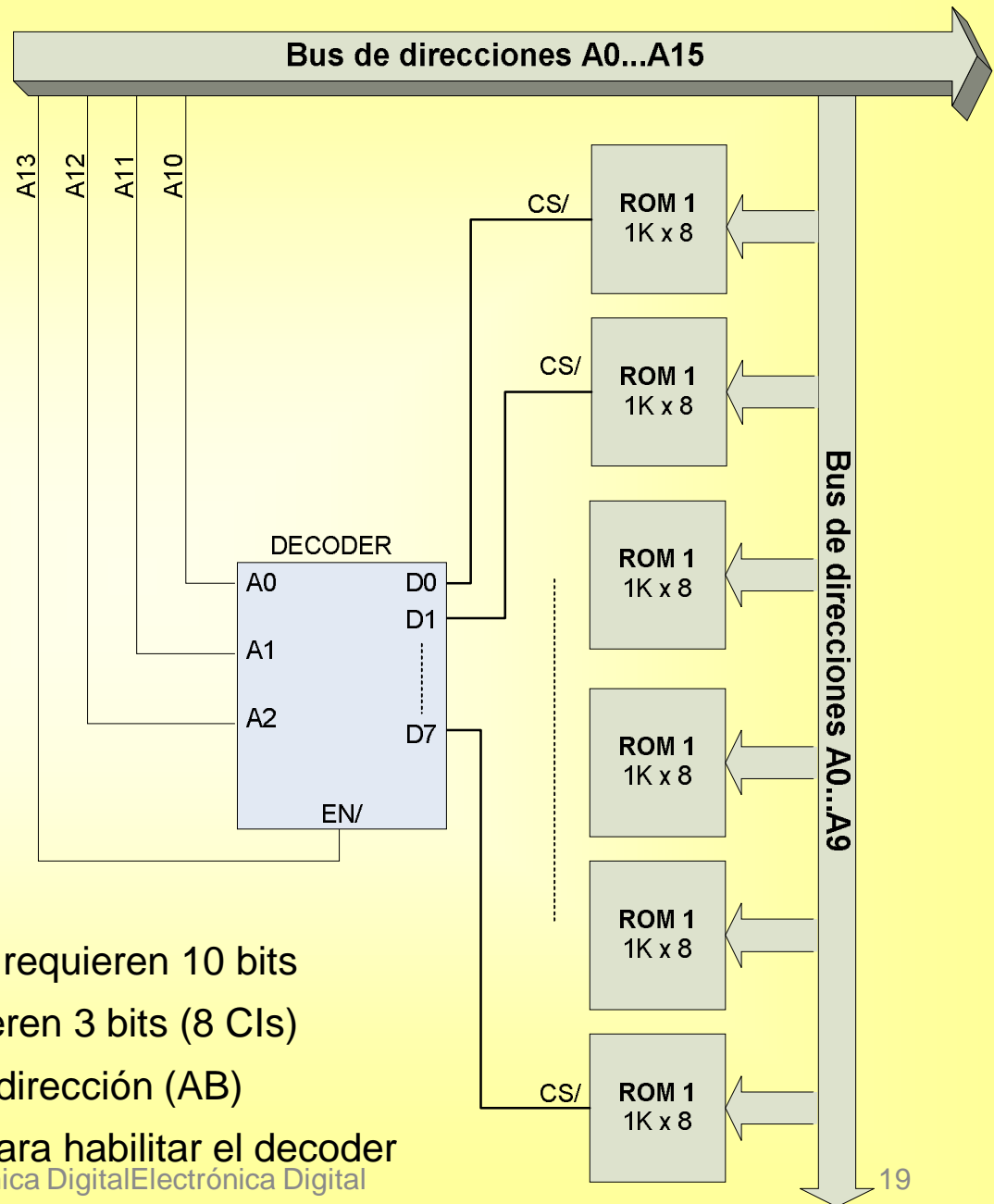
Bits de selección						Bits de dirección							
A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₀	Rango	Bytes	CI
0	0	0	0	0	0	0	0	0	...	0	0000 03FF	1K	No usados
0	0	0	0	0	0	1	1	1	...	1			
0	0	0	0	0	1	0	0	0	...	0	0400 07FF	1K	ROM #5
0	0	0	0	0	1	1	1	1	...	1			
0	0	0	0	1	0	0	0	0	...	0	0800 0BFF	1K	ROM #4
0	0	0	0	1	0	1	1	1	...	1			
0	0	0	0	1	1	0	0	0	...	0	0C00 0FFF	1K	No usados
0	0	0	0	1	1	1	1	1	...	1			
0	0	0	1	0	0	0	0	0	...	0	1000 13FF	1K	ROM #3
0	0	0	1	0	0	1	1	1	...	1			
0	0	0	1	0	1	0	0	0		0	1400 1FFF	3K	No usados
0	0	0	1	1	1	1	1	1	...	1			
0	0	1	0	0	0	0	0	0	...	0	2000 23FF	1K	ROM #2
0	0	1	0	0	0	1	1	1	...	1			

Mapa de memoria (cont.)

Bits de selección						Bits de dirección					Rango	Bytes	CI
A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₀			
0	0	1	0	0	1	0	0	0	...	0	2400 3FFF	7K	No usados
0	0	1	1	1	1	1	1	1	...	1			
0	1	0	0	0	0	0	0	0	...	0	4000 43FF	1K	ROM#1
0	1	0	0	0	0	1	1	1	...	1			
0	1	0	0	0	1	0	0	0	0	4400 7FFF	15K	No usados
0	1	1	1	1	1	1	1	1	1			
1	0	0	0	0	0	0	0	0	0	8000 81FF	512	RAM
1	0	0	0	0	0	0	1	1	1			
1	0	0	0	0	0	1	0	0	...	0	8200 83FF	512	RAM (espejo)
1	0	0	0	0	0	1	1	1	...	1			
1	0	0	0	0	1	0	0	0	...	0	8400 FFFF	31K	No usados
1	1	1	1	1	1	1	1	1	...	1			

Selección decodificada

Caso similar al anterior con direccionamiento de 8 bloques de 1K x 8 usando un *decoder*



- Para direccionar las RAM y ROM se requieren 10 bits
- Para seleccionar (decoder) se requieren 3 bits (8 CIs)
- Se requiere un total de 13 líneas de dirección (AB)
- Se puede agregar un bit extra (14) para habilitar el decoder

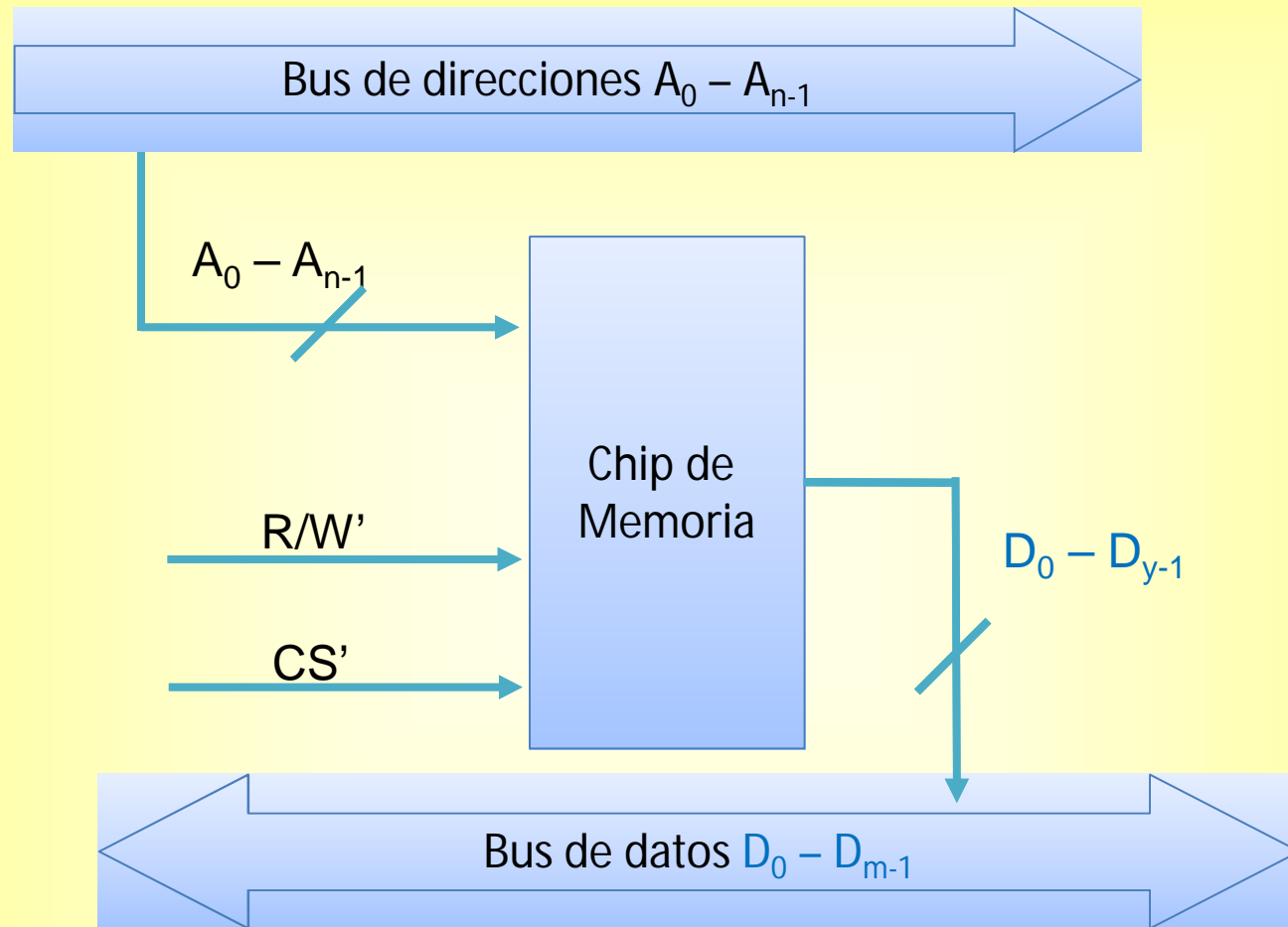
Mapa de memoria

DECODER (activo por L)											
EN	A2	A1	A0								
	Bits de selección			Direcciones físicas (10 bits)							
A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	...	A ₀	Decoder	Direcciones	CI
0	0	0	0	-	-	-	...	-	D0	0000 - 03FF	ROM0
	0	0	1	-	-	-	...	-	D1	0400 - 07FF	ROM1
	0	1	0	-	-	-	...	-	D2	0800 - 0BFF	ROM2
	0	1	1	-	-	-	...	-	D3	0CFF - 0FFF	ROM3
	1	0	0	-	-	-	...	-	D4	1000 - 13FF	ROM4
	1	0	1	-	-	-	...	-	D5	1400 - 17FF	ROM5
	1	1	0	-	-	-	...	-	D6	1800 - 1BFF	ROM6
	1	1	1	-	-	-	...	-	D7	1C00 - 1FFF	ROM7

Aumento de la capacidad: bancos de memoria

- **Expansión de la longitud de la palabra dato**
 - Ejemplo: banco de $1K \times 8$ con memorias de $1K \times 4$
- **Expansión de la capacidad de almacenamiento**
 - Ejemplo: banco de $2K \times 4$ con memorias de $1K \times 4$
- **Expansión de la capacidad y la longitud de la palabra**
 - Ejemplo: banco de $2K \times 8$ con memorias de $1K \times 4$

Expansión del tamaño de la palabra dato



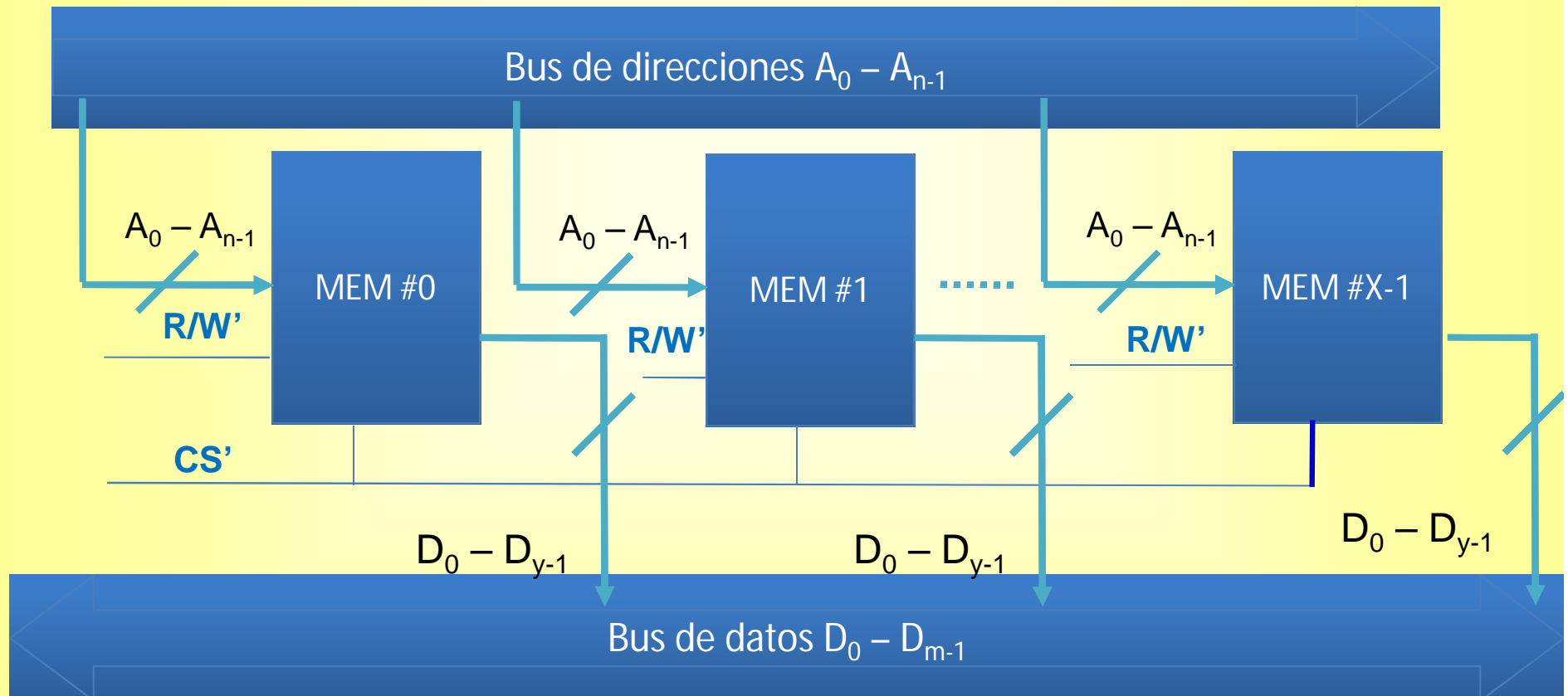
m: número de bits de longitud del nuevo dato

y: número de bits de longitud del dato de cada memoria

→ $m / y = X$ memorias necesarias

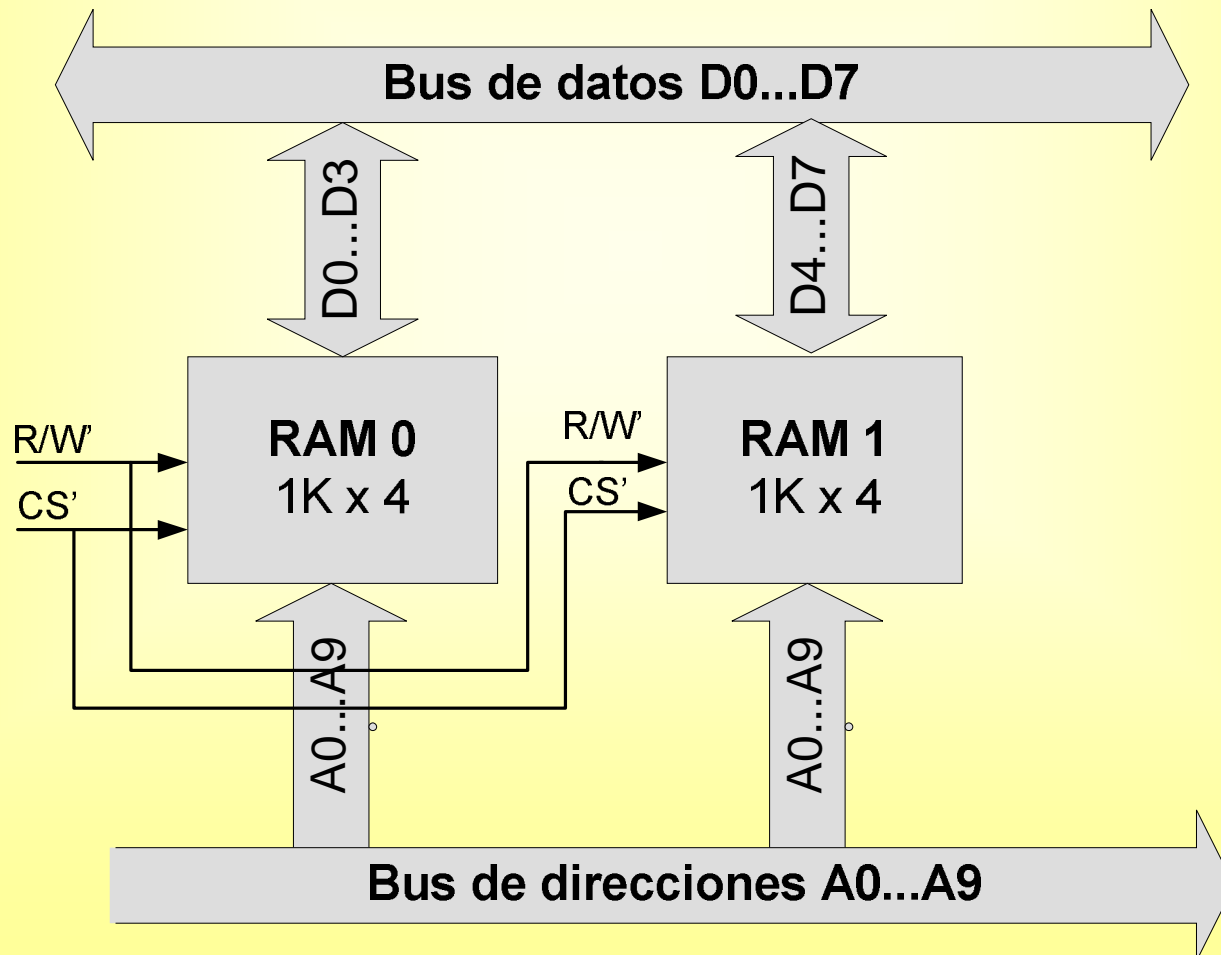
Arquitectura genérica

Las líneas de direcciones entran en paralelo a todas las memorias
Las líneas de control CS' y R/W' están conectadas en paralelo a cada memoria



Ejemplo: banco de 1K x 8 con memorias de 1K x 4

- #bits del AB de cada memoria? 10
- #bits del DB de cada memoria? $y = 4$
- #bits del DB del banco? $m = 8$
- # de memorias de 1K x 4? $x = m/y = 8/4 = 2$



Expansión de la capacidad de almacenamiento

$$N = 2^n$$

N es la capacidad inicial de la memoria
(**n** es el número de bits del bus de direcciones)

$$M = 2^z$$

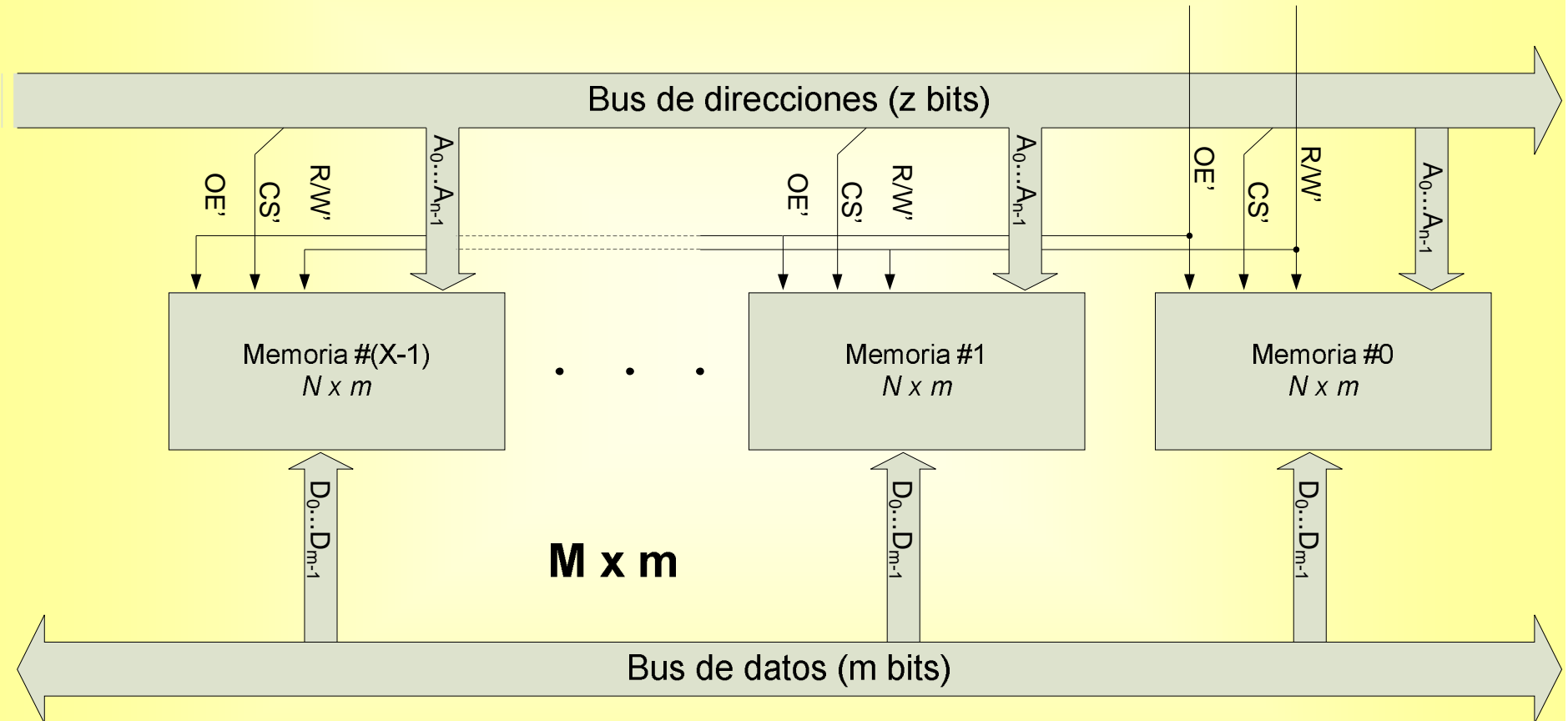
M es la capacidad final de la memoria (banco)
(**z** el número de bits del **nuevo** bus de direcciones)

$$\frac{M}{N} = \frac{2^z}{2^n} = 2^{z-n} = X$$

Número de memorias necesarias

Arquitectura genérica

Las nuevas líneas de dirección permiten la operación de cada una de las memorias actuando sobre el CS/



Ejemplo

Banco RAM de 4K x 4 con memorias de 1K x 4 (**selección lineal**)

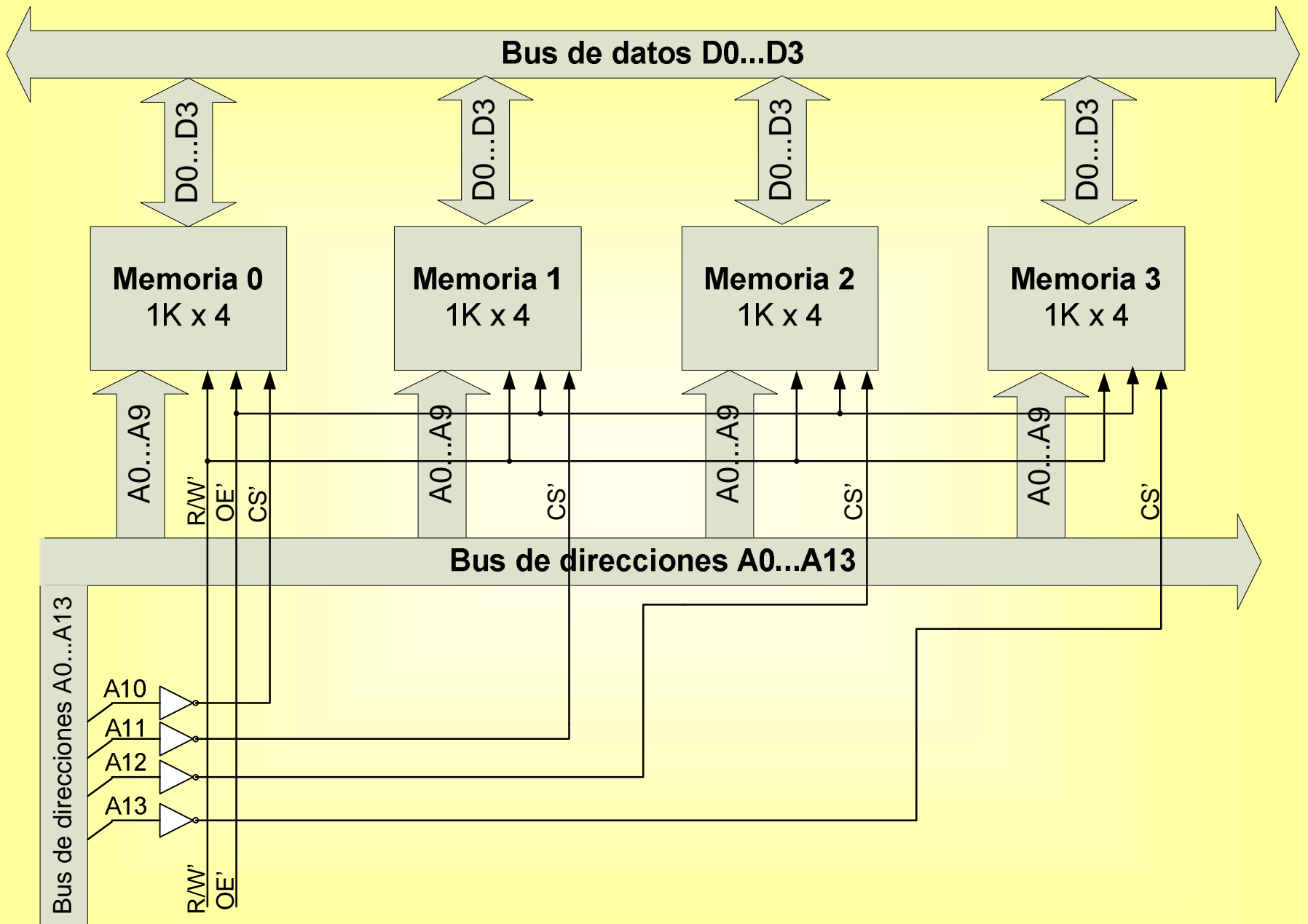
Selección lineal → se necesitan tantas líneas de dirección adicionales como memorias contenga el banco

Capacidad inicial $N = 1K = 1024$
AB: $n = 10$ bits ($A_0 \dots A_9$)

Capacidad final $M = 4K = 4096$

Cantidad de memorias
 $X = M / N = 4$ memorias

Nuevas líneas de dirección: 4 ($A_{10}, A_{11}, A_{12}, A_{13}$) (selección lineal)
AB = 14 bits ($A_0 \dots A_{13}$)



Mapa de memoria

Dir. hexa	Direccionamiento (Selección de chip)				Direccionamiento físico (de cada chip)										Chip
	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	
0400 a 07FF	0	0	0	1	0	0	0	0	0	0	0	0	0	0	RAM0
	0	0	0	1	1	1	1	1	1	1	1	1	1	1	
0800 a 0BFF	0	0	1	0	0	0	0	0	0	0	0	0	0	0	RAM1
	0	0	1	0	1	1	1	1	1	1	1	1	1	1	
1000 a 13FF	0	1	0	0	0	0	0	0	0	0	0	0	0	0	RAM2
	0	1	0	0	1	1	1	1	1	1	1	1	1	1	
2000 a 23FF	1	0	0	0	0	0	0	0	0	0	0	0	0	0	RAM3
	1	0	0	0	1	1	1	1	1	1	1	1	1	1	

Existen partes no usadas
del espacio de direcciones:

0000 a 03FF

0C00 a 0FFF

1400 a 1FFF

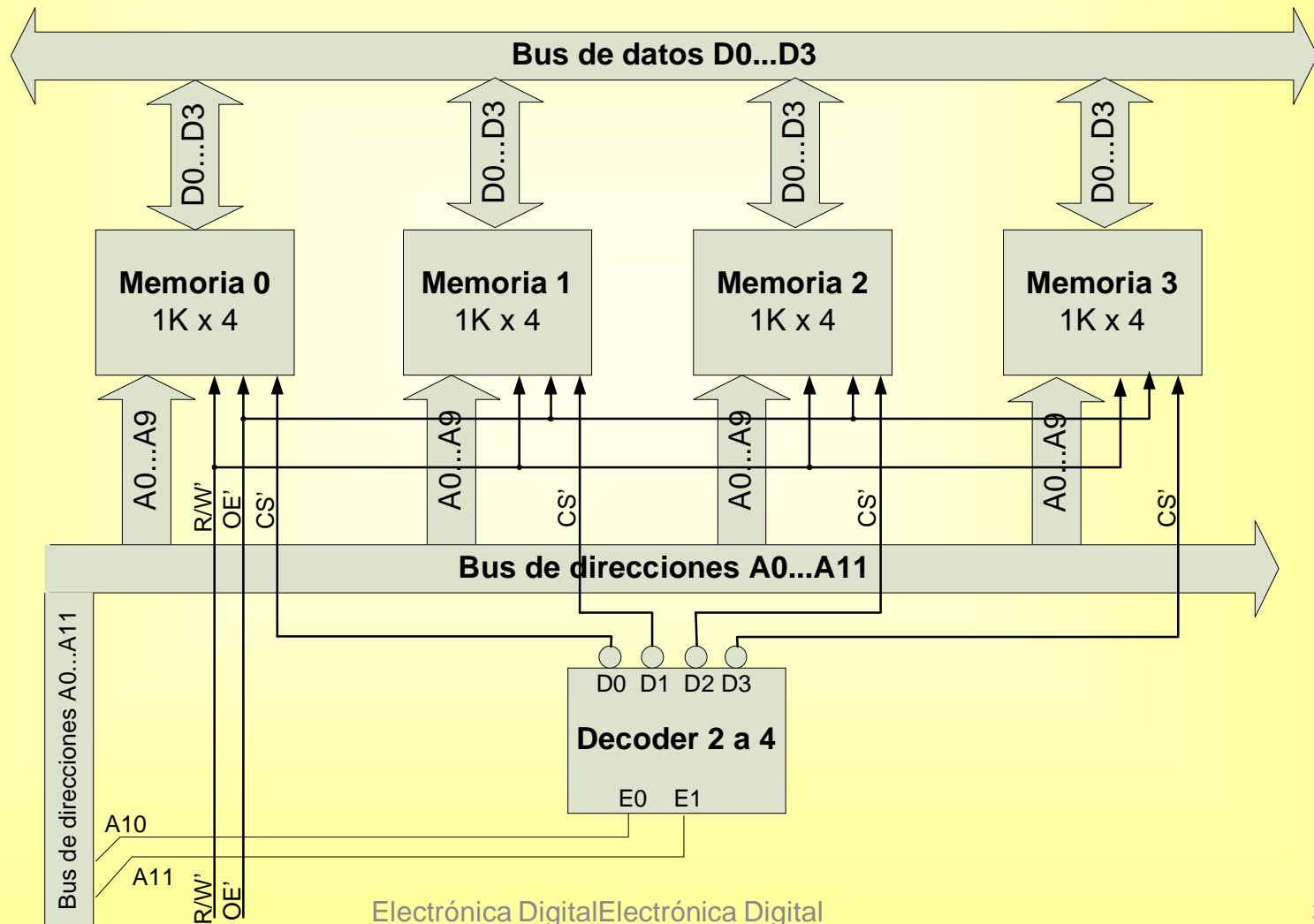
2400 a 3FFF

Ejemplo:

Banco RAM de 4K x 4 con memorias de 1K x 4 (selección decodificada)

- Nuevas líneas de dirección

$$m - n = 2 \quad (A_{10}, A_{11})$$



Mapa de memoria

Direc. Hexa	Direc. Decimal	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	Chip
		Selección		Direcciones para cada chip										
000 a 3FF	0 a 1023	0	0	0	0	0	0	0	0	0	0	0	0	RAM0
		0	0	1	1	1	1	1	1	1	1	1	1	
400 a 7FF	1024 a 2047	0	1	0	0	0	0	0	0	0	0	0	0	RAM1
		0	1	1	1	1	1	1	1	1	1	1	1	
800 a BFF	2048 a 3071	1	0	0	0	0	0	0	0	0	0	0	0	RAM2
		1	0	1	1	1	1	1	1	1	1	1	1	
C00 a FFF	3072 a 4095	1	1	0	0	0	0	0	0	0	0	0	0	RAM3
		1	1	1	1	1	1	1	1	1	1	1	1	

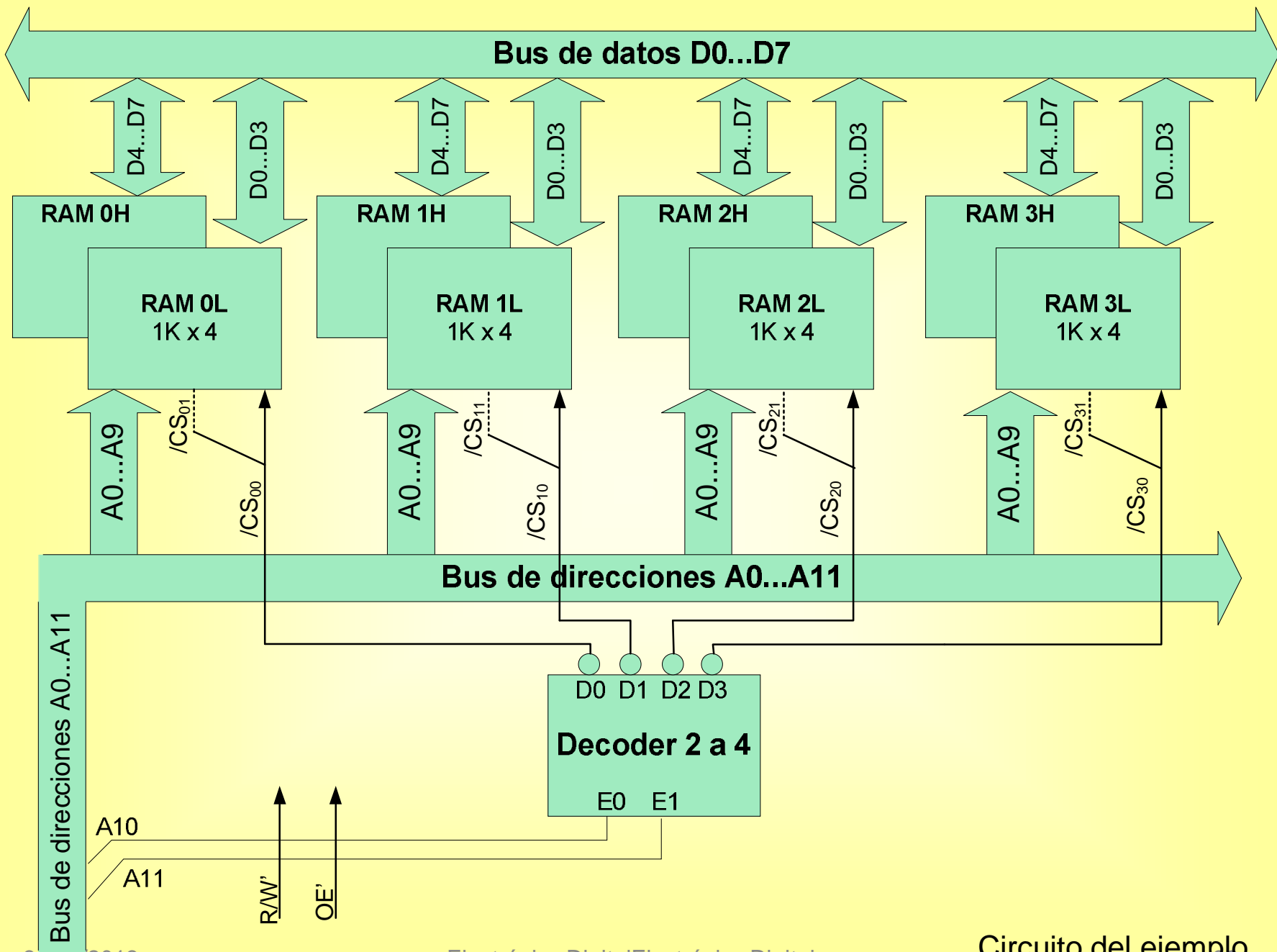
A ₁₁	A ₁₀	decoder	Dir. Hex	Dir. Decim.	Memoria
0	0	D0	000 a 3FF	0 a 1023	RAM 0
0	1	D1	400 a 7FF	1024 a 2047	RAM 1
1	0	D2	800 a BFF	2048 a 3071	RAM 2
1	1	D3	C00 a FFF	3072 a 4095	RAM 3

El mapa se aprovecha por completo en forma lineal

Expansión de longitud de dato y de la capacidad total

Ejemplo: banco RAM de 4K x 8 con memorias de 1K x 4

- Capacidad inicial $N = 1K = 1024$
 $n = 10$ bits ($A_0 \dots A_9$)
- Capacidad final $M = 4K = 4096$
 $m = 12$ bits ($A_0 \dots A_{11}$)
- Cantidad de memorias
 $X = M / N = 4$ memorias \rightarrow **8 memorias**
- Nuevas líneas de dirección (mínimo): $12 - 10 = 2$ (A_{10}, A_{11})



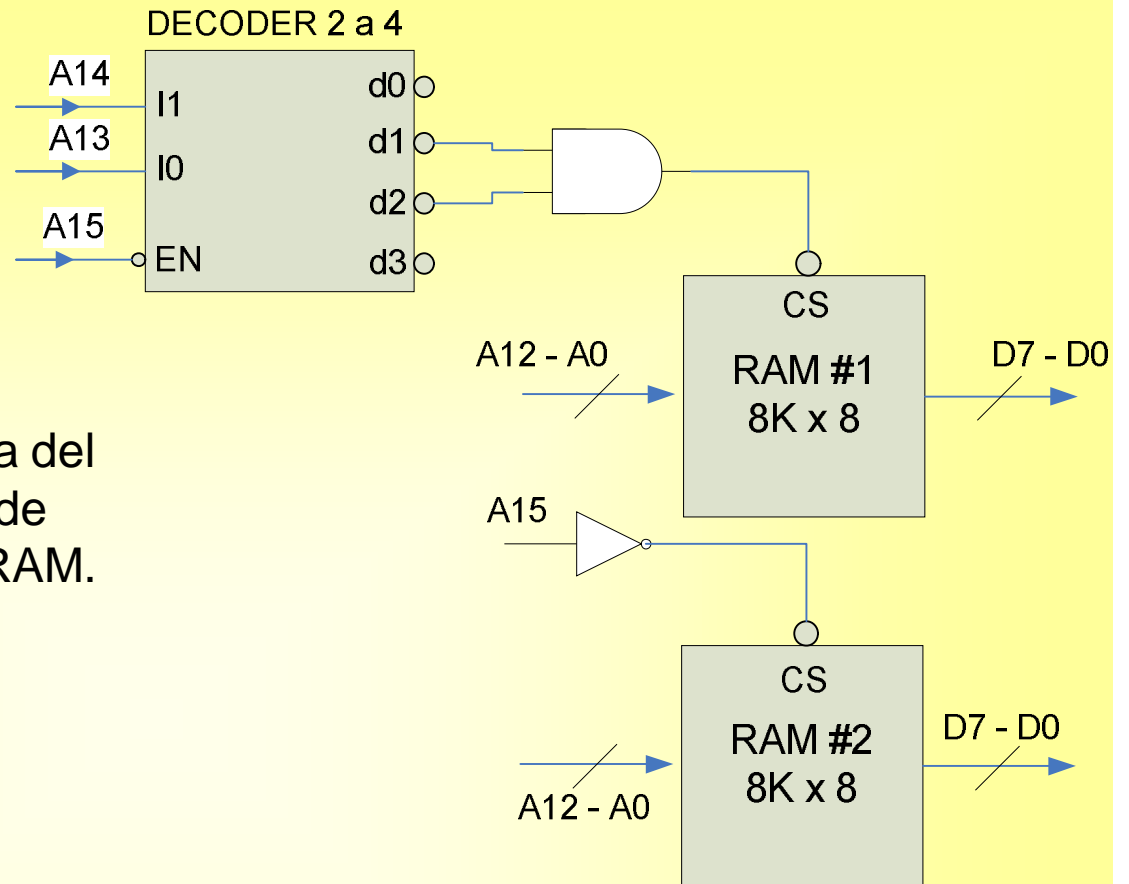
Mapa de memoria

Direc. Hexa	Direc. Dec.	Selección		Direccionamiento										Chip
		A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	
000 a 3FF	0 a 1023	0	0	0	0	0	0	0	0	0	0	0	0	RAM0L RAM0H
		0	0	1	1	1	1	1	1	1	1	1	1	
400 a 7FF	1024 a 2047	0	1	0	0	0	0	0	0	0	0	0	0	RAM1L RAM1H
		0	1	1	1	1	1	1	1	1	1	1	1	
800 a BFF	2048 a 3071	1	0	0	0	0	0	0	0	0	0	0	0	RAM2L RAM2H
		1	0	1	1	1	1	1	1	1	1	1	1	
C00 a FFF	3072 a 4095	1	1	0	0	0	0	0	0	0	0	0	0	RAM3L RAM3H
		1	1	1	1	1	1	1	1	1	1	1	1	

Dir. Hex	Dir. Decim.	A ₁₁	A ₁₀	Memoria
000 a 3FF	0 a 1023	0	0	#0
400 a 7FF	1024 a 2047	0	1	#1
800 a BFF	2048 a 3071	1	0	#2
C00 a FFF	3072 a 4095	1	1	#3

‘Espejos’ en el mapa de memoria

Caso: Hallar el mapa de memoria del circuito indicando las posiciones de memoria ocupadas por cada CI RAM.



A15	A14	A13	Decoder	Mem	Posición en el mapa	Bytes
0	0	0	d0	-	H0000 – H1FFF	8K
	0	1	d1	RAM1	H2000 – H3FFF	8K
	1	0	d2	RAM1	H4000 – H5FFF	8K
	1	1	d3	-	H6000 – H7FFF	8K
1	X	X		RAM2	H8000 - HFFFF	32K

0	0000	No usado
	
8191	1FFF	RAM #1
8192	2000	
	
16383	3FFF	RAM #1 (espejo)
16384	4000	
	
24575	5FFF	No usado
40960	6000	
	
49151	7FFF	RAM #2
41152	8000	
	
65535	FFFF	

Por ejemplo, a la primer posición de memoria de RAM #2 se accede indistintamente con las direcciones:

	$A_{15} \dots A_{12}$	$A_{11} \dots A_0$
8000	1 0 0 0	0000 0000 0000
A000	1 0 1 0	0000 0000 0000
C000	1 1 0 0	0000 0000 0000
E000	1 1 1 0	0000 0000 0000

