

# Clase 5

Práctico de Diseño Lógico

# Clase 5 – temas

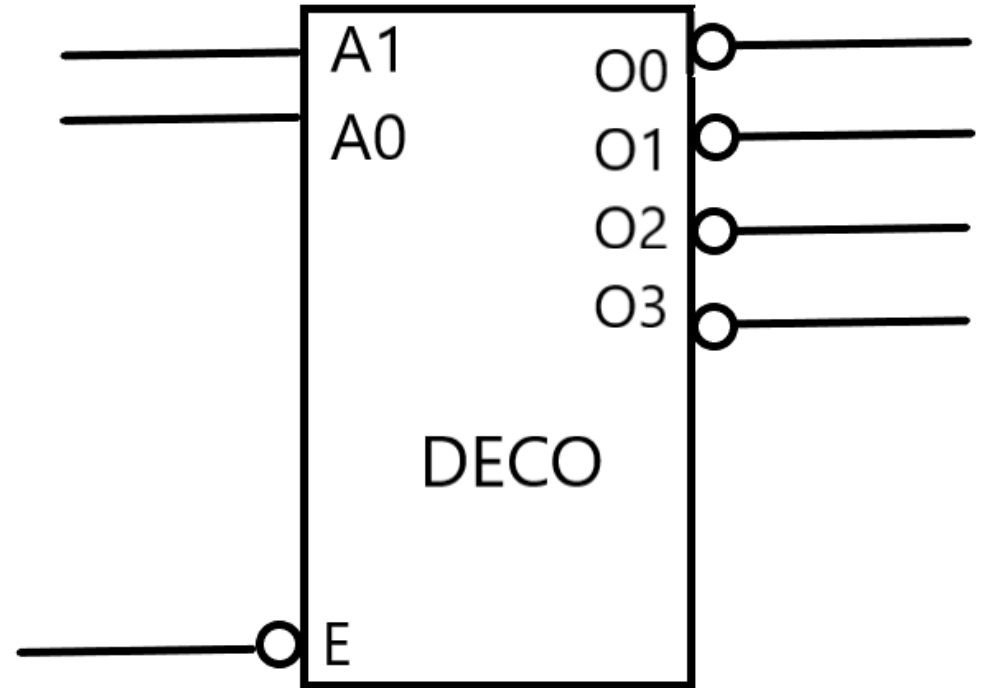
- Dispositivos MSI
  - MUX
  - EPROM
- Retardos

# Repaso Dispositivos MSI

- Dispositivos MSI (Medium Scale Integration). Son dispositivos que integran cosas simples (unidades de memoria, funciones lógicas básicas) para dar funcionalidad más compleja.
- Algunos ejemplos:
  - Decodificador
  - Multiplexor
  - Memorias
  - compuertas Tri-state
- **Siempre indicar nombres (poner etiquetas),** no dibujar un rectángulo con rayitas.

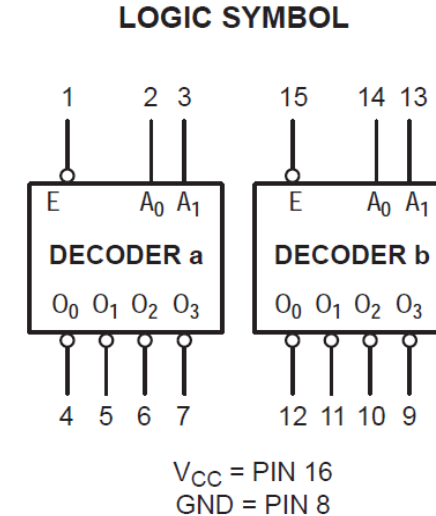
# Decodificador 1 a 4

- Cuando está deshabilitado ( $E=1$ ), las cuatro salidas ( $O_0, O_1, O_2, O_3$ ) están en 1.
- Cuando está habilitado ( $E=0$ ), la salida activa en bajo es la de la posición  $A_1A_0$ , el resto quedan inactivas en 1.



# Decodificador – SN74LS139

- Este chip trae dos deco de 1 a 4.
- Cuando está deshabilitado ( $E=1$ ), las cuatro salidas ( $O_0, O_1, O_2, O_3$ ) están en 1.
- Cuando está habilitado ( $E=0$ ), la salida activa en bajo es la de la posición  $A_1A_0$ , el resto quedan inactivas en 1.
- El diagrama lógico tiene una compuerta NAND por cada salida.
- También puede ser utilizado como demultiplexor.



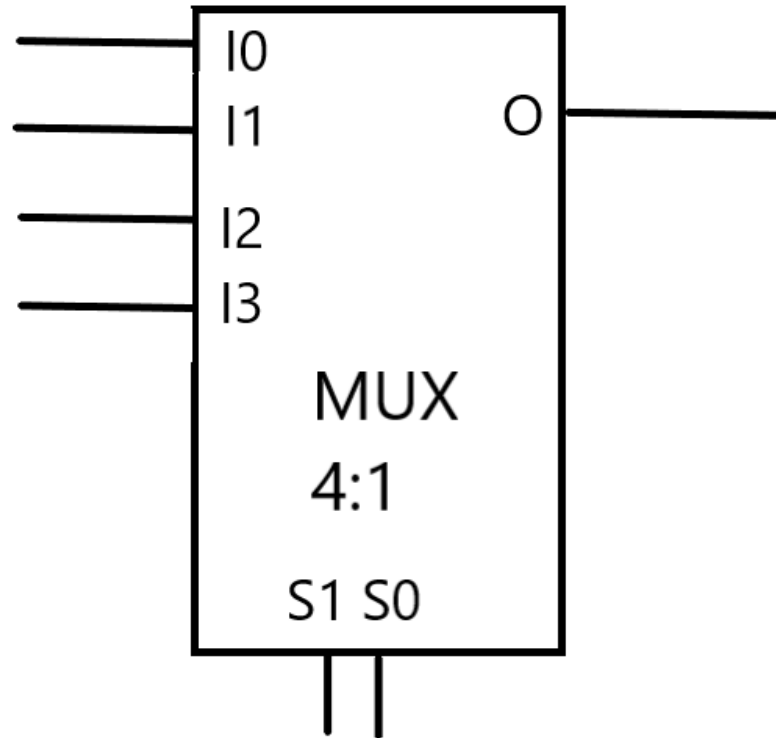
**TRUTH TABLE**

INPUTS			OUTPUTS			
$\bar{E}$	$A_0$	$A_1$	$\bar{O}_0$	$\bar{O}_1$	$\bar{O}_2$	$\bar{O}_3$
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L

H = HIGH Voltage Level  
L = LOW Voltage Level  
X = Don't Care

# Multiplexor - SN74LS153

- Este chip trae dos MUX de 4 entradas a 1 salida.
- Cada MUX tiene 2 patas de selección ( $S_1$  y  $S_0$ ).
- La salida O toma el valor de la entrada seleccionada por  $S_1$  y  $S_0$ .

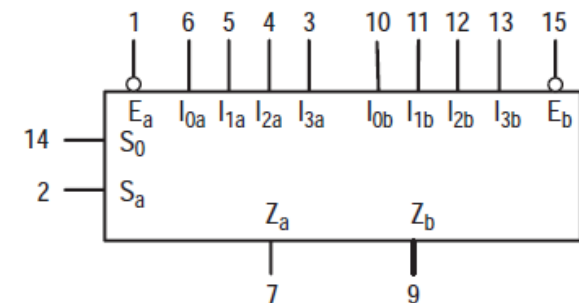


TRUTH TABLE

SELECT INPUTS		$\bar{E}$	INPUTS (a or b)				OUTPUT Z
$S_0$	$S_1$		$I_0$	$I_1$	$I_2$	$I_3$	
X	X	H	X	X	X	X	L
L	L	L	L	X	X	X	L
L	L	L	H	X	X	X	H
H	L	L	X	L	X	X	L
H	L	L	X	H	X	X	H
L	H	L	X	X	L	X	L
L	H	L	X	X	H	X	H
H	H	L	X	X	X	L	L
H	H	L	X	X	X	H	H

H = HIGH Voltage Level  
L = LOW Voltage Level

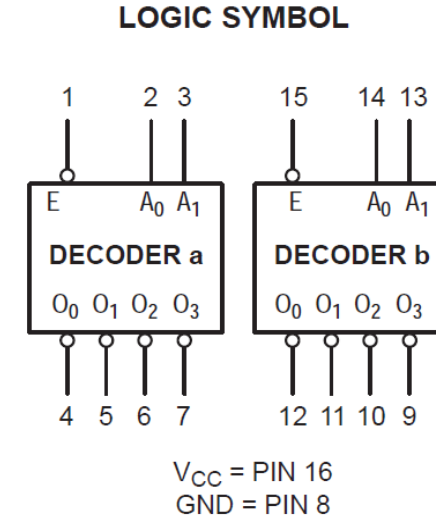
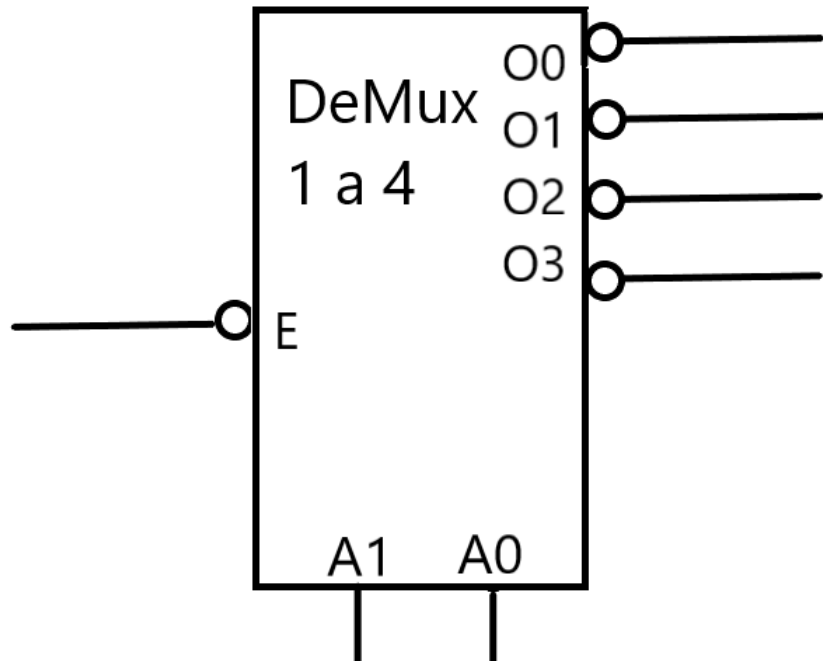
LOGIC SYMBOL



$V_{CC}$  = PIN 16  
GND = PIN 8

# Demultiplexor - SN74LS139

- El SN74LS139 se puede usar como demultiplexor con entrada E.



TRUTH TABLE

INPUTS			OUTPUTS			
$\bar{E}$	$A_0$	$A_1$	$\bar{O}_0$	$\bar{O}_1$	$\bar{O}_2$	$\bar{O}_3$
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L

H = HIGH Voltage Level  
 L = LOW Voltage Level  
 X = Don't Care

# Ejercicio 5.4

**Ejercicio 4.** Emplear un multiplexor para implementar la siguiente ecuación:

$$y = !d \ c \ !b \ a + !d \ !c \ b \ a + d \ !c \ b \ !a + d \ c \ !b \ a + !d \ !c \ !b \ !a + d \ c \ !b \ !a + d \ c \ b \ a + d \ !c \ !b \ a$$

- a) Utilizar un multiplexor de 16 a 1.
- b) Utilizar un multiplexor de 8 a 1 e inversores.

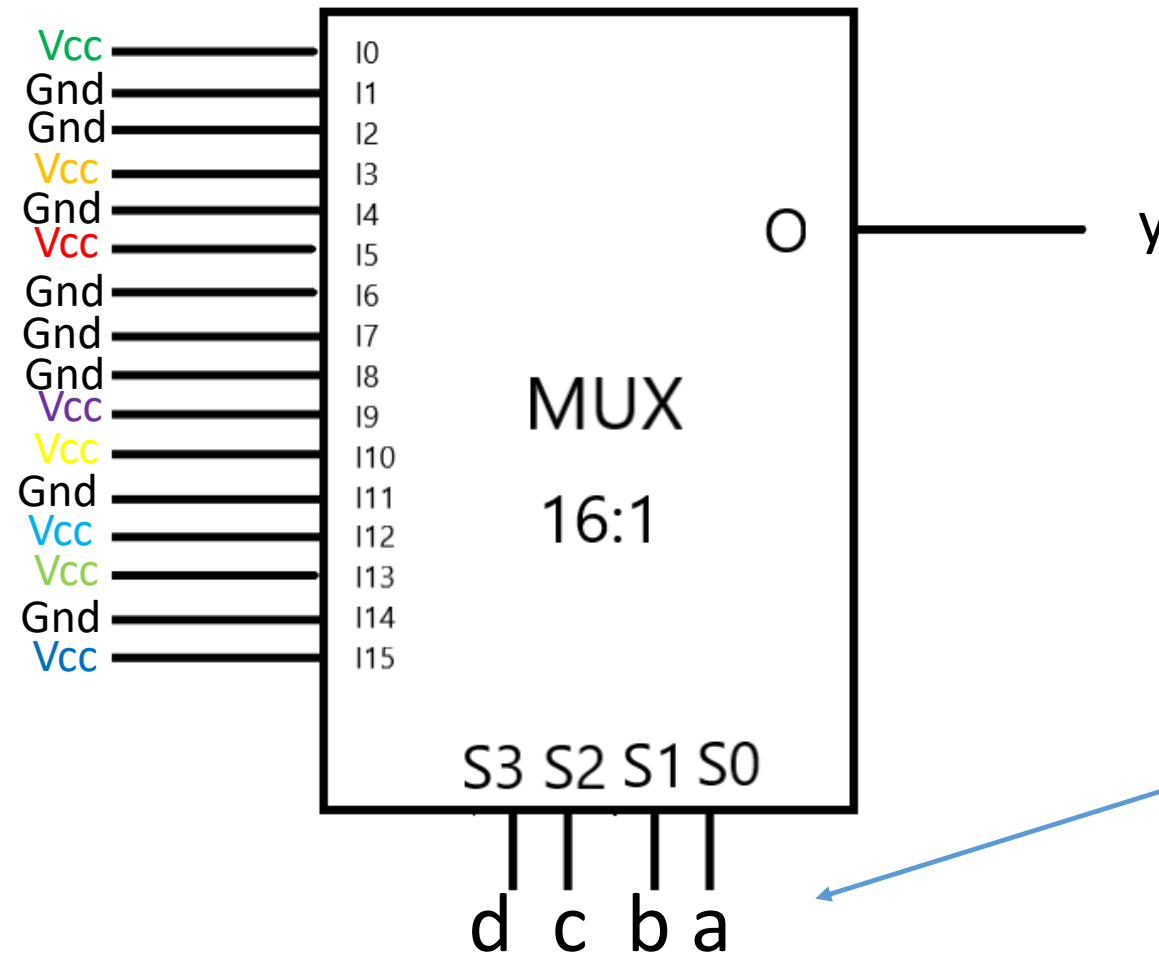
- Un multiplexor de 16 a 1 tiene 16 entradas [ $I_{15}..I_0$ ].
- Necesito 4 pines [ $S_3..S_0$ ] de selección para elegir cuál de esas 16 entradas se conecta a la salida.
- La función **y** tomará el valor 1 cuando alguno de los 8 minitérminos de la suma sean 1.
- Puedo conectar las entradas **d c b a** a las patas de selección **S<sub>3</sub> S<sub>2</sub> S<sub>1</sub> S<sub>0</sub>**.
- ¿Qué valores debo conectar en las entradas [ $I_{15}..I_0$ ]?



# Ejercicio 5.4

$$y = \boxed{\neg d c \neg b a} + \boxed{\neg d \neg c b a} + \boxed{d \neg c b \neg a} + \boxed{d c \neg b a} + \boxed{\neg d \neg c \neg b \neg a} + \boxed{d c \neg b \neg a} + \boxed{d c b a} + \boxed{d \neg c \neg b a}$$

0101



Conecto las variables de la función a los pines de selección

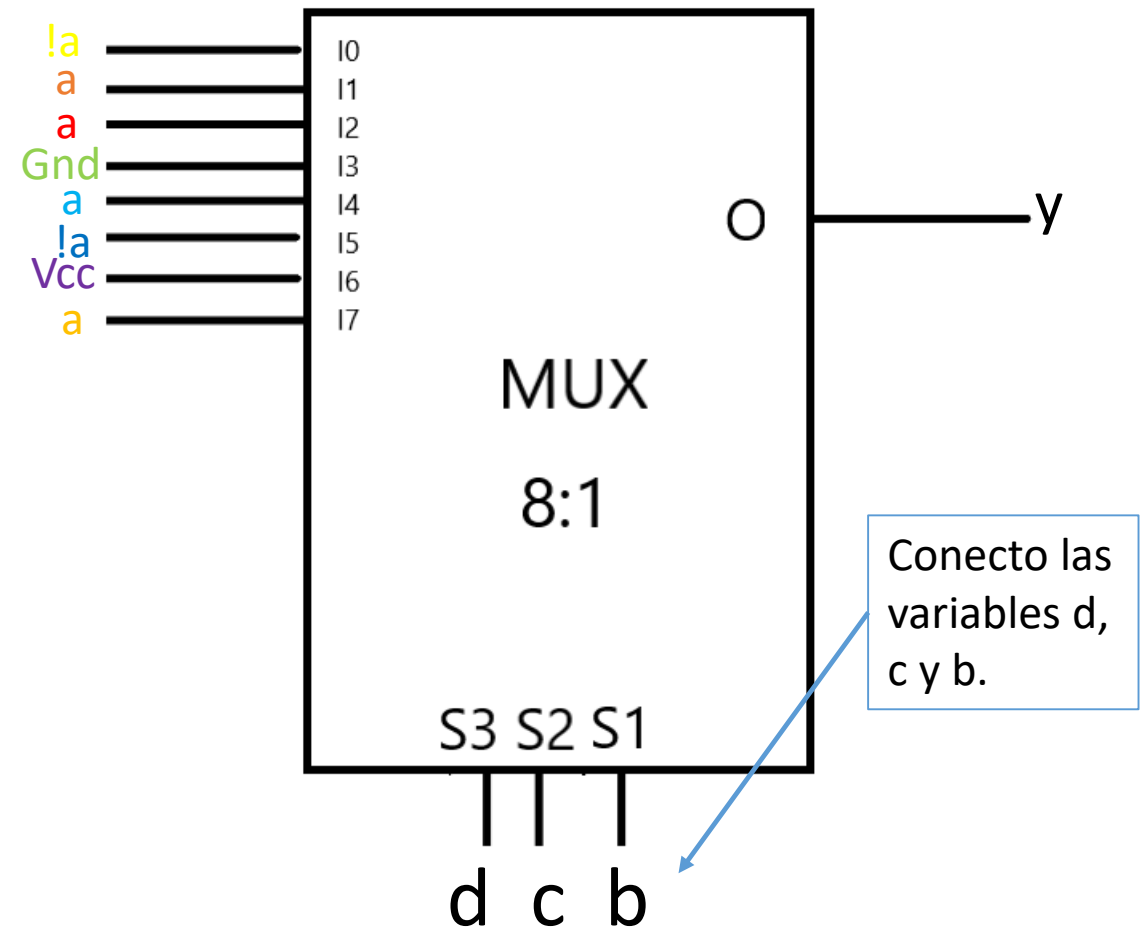
$$y = !d c !b a + !d !c b a + d !c b !a + d c !b a + !d !c !b !a + d c !b !a + d c b a + d !c !b a$$

## Parte b)

- Ahora solo tenemos un multiplexor de 8 a 1 e inversores.

d	c	b	a	y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

d	c	b	a	y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1



# Repaso EPROM

- Podemos imaginar la ROM como una tabla, en cada dirección uno selecciona una celda (palabra) que contiene bits.
- Con N bits de direcciones podemos seleccionar  $2^n$  celdas diferentes.
- Comúnmente se indica que una memoria es de  $XK * M$ . Con esto se indica que hay  $X \times 2^{10}$  celdas y que cada celda tiene un ancho de M bits.

# Ejercicio 8 de Práctico repaso para parcial 1.

## **Ejercicio 8** (Ex. Enero 1996)

Se desea implementar un comparador de números A y B de 8 bits en complemento a 2. Para ello se dispone de una EPROM de 8 k x 8 bits y multiplexores.

- Dar el circuito para generar una función MAYOR de 1 bit que indique si A es mayor que B. Especificar claramente las conexiones a la EPROM.
- Utilizando la función generada en (a) realizar un circuito que conecte a la salida el mayor de los dos números.
- Indicar para los siguientes casos dónde se almacena en la EPROM el resultado de la función MAYOR y cuánto vale.

A = 0011 1111

B = 1111 1001

A = 0100 0000

B = 0011 1111

- EPROM 8K x 8
- $8K = 2^3 \times 2^{10} = 2^{13} \Rightarrow$  necesito 13 bits para direccionar la memoria.
- Con un multiplexor de 8 a 1 puedo elegir el bit de la palabra de datos seleccionada  $\Rightarrow$  necesito 3 bits para la selección (S[2..0]).
- Conecto A[7..0] a Add[12..5], B[7..3] a Add[4..0] y B[2..0] a S[2..0].

# Ejercicio 8 parte b)

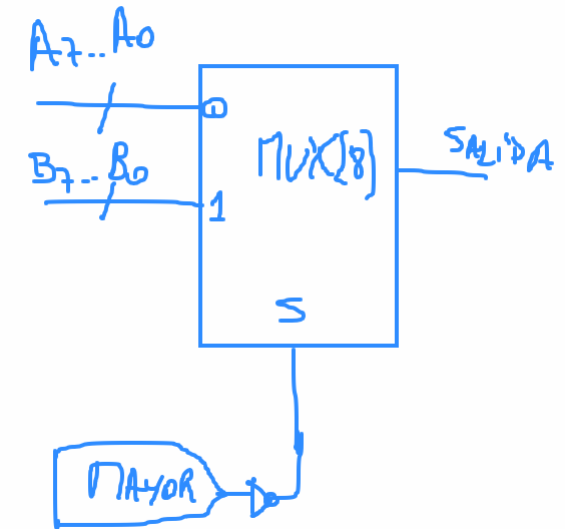
## Ejercicio 8 (Ex. Enero 1996)

Se desea implementar un comparador de números A y B de 8 bits en complemento a 2. Para ello se dispone de una EPROM de 8 k x 8 bits y multiplexores.

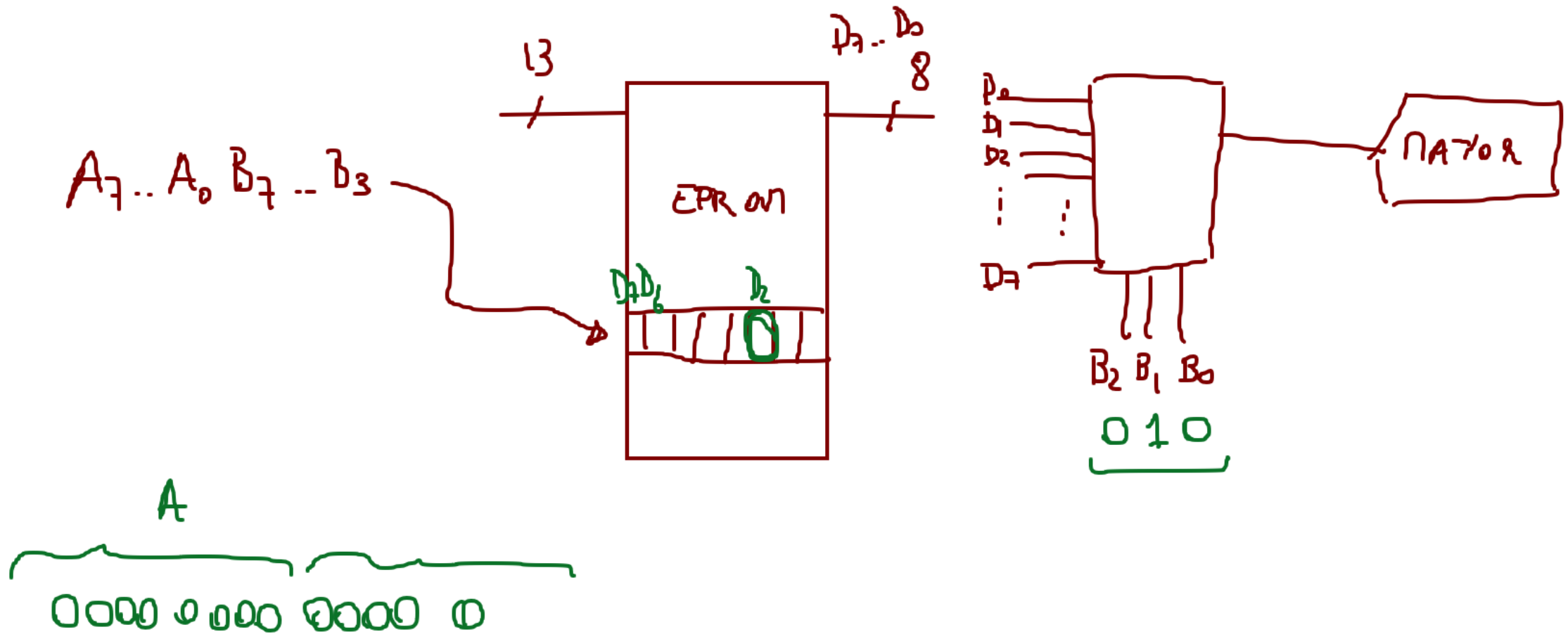
- Dar el circuito para generar una función MAYOR de 1 bit que indique si A es mayor que B. Especificar claramente las conexiones a la EPROM.
- Utilizando la función generada en (a) realizar un circuito que conecte a la salida el mayor de los dos números.
- Indicar para los siguientes casos dónde se almacena en la EPROM el resultado de la función MAYOR y cuánto vale.

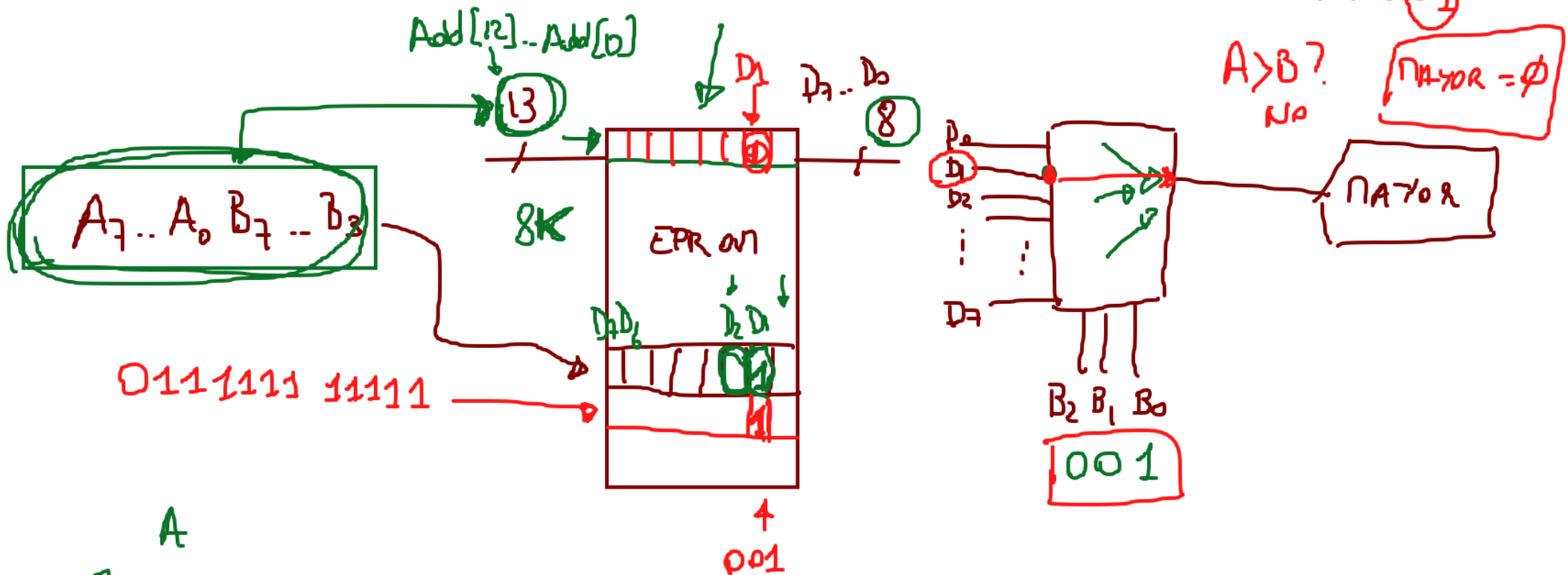
A = 0011 1111  
B = 1111 1001

A = 0100 0000  
B = 0011 1111



- Puedo utilizar un multiplexor de 2 a 1 de ancho 8 para conectar a la salida el mayor de los dos números, para eso conecto a la pata de selección del MUX la señal MAYOR.





0111111 11111

A

0000 0000 0000 0

0011 1111 1111 1

A

B

001

$\Pi A70R = 1$

$Add[12] \quad Add[11] \quad Add[10] \quad \dots \quad Add[0]$   
 $A_7 \quad A_6 \quad A_5 \quad \dots \quad B_3$

# Tiempos

- Las señales no cambian instantáneamente. El fabricante dice que una salida puede cambiar entre 0 y  $n$  nanosegundos después de un cambio en la entrada.