

CIRCUITOS COMBINACIONALES

CIRCUITOS COMBINACIONALES.

1.- Circuitos combinacionales M.S.I.

2.- Decodificadores.

Decodificador de 2 entradas.

Decodificador Octal.

Decodificador BCD – Decimal.

Decodificador Hexadecimal.

3.- Codificadores.

Codificador sin prioridad.

Codificador con prioridad.

Codificador de prioridad octal – binario (74148).

4.- Multiplexores.

Conexión de multiplexores.

5.- Demultiplexores.

Demultiplexor a partir de un decodificador.

Demultiplexor funcionando como decodificador.

Utilización de multiplexores y demultiplexores en la transmisión de datos por una línea.

6.- Convertidores de código. Decodificador BCD a 7 segmentos.

7.- Comparadores.

Comparador de 1 bit.

Comparador de 3 bit.

Comparador de 4 bit (7485).

Comparador de dos números de 5 bit.

Comparador de dos números de 8 bit.

Circuitos combinacionales M.S.I.

Un circuito combinacional es aquel en el que la salida depende en todo momento de los valores de las variables de entrada.

Un circuito combinacional realiza una función lógica.

Estos circuitos están formados por varias puertas conectadas entre sí y agrupadas en un solo integrado MSI (Media escala de integración)

Se utilizan para reducir el tamaño del circuito impreso al poder utilizar menos componentes, reducir el tiempo de diseño y aumentar la inmunidad al ruido al disminuir el número de conexiones.

EJEMPLOS DE CIRCUITOS COMBINACIONALES M.S.I.

DECODIFICADORES

CODIFICADORES

MULTIPLEXORES

DEMULTIPLEXORES

CONVERTIDORES DE CÓDIGO

COMPARADORES

Decodificadores.

- Son circuitos combinacionales de **N entradas** y un número de **salidas $\leq 2^N$** .
- La función básica de un decodificador es detectar la presencia de una determinada combinación de bits (código) en sus entradas e indicar la presencia de ese código activando una determinada salida.
- Para cada combinación binaria de sus entradas, se activa **una sola** de sus salidas.
- La salida que se activa numerada en decimal, corresponde a la combinación binaria de los bit de entrada.

TIPOS DE DECODIFICADORES.

- ❑ **EXCITADORES.** Son aquellos en los que en sus salidas disponen de potencia suficiente para activar elementos indicadores como diodos Led, etc.
- ❑ **NO EXCITADORES.** Tienen corrientes de salida bajas, por lo que sólo se pueden acoplar a otros circuitos digitales.

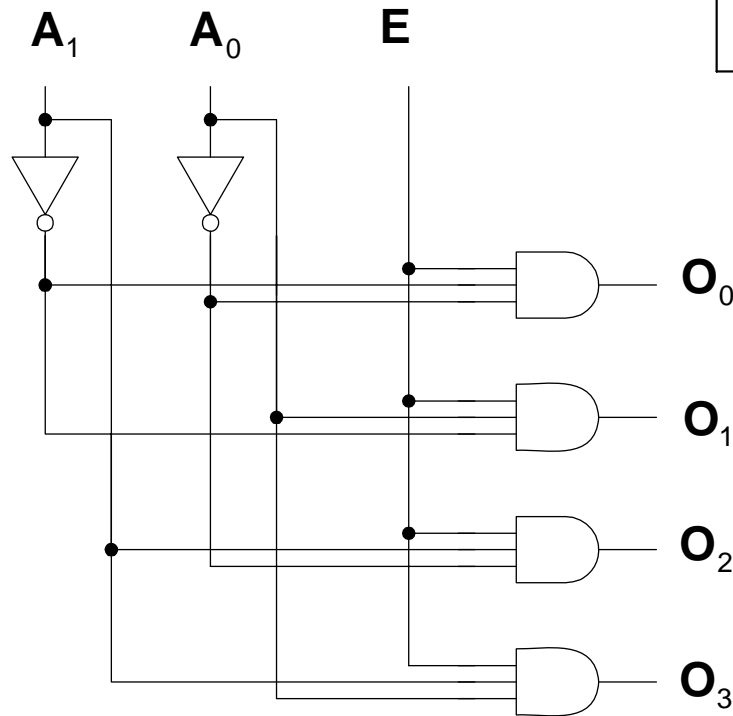
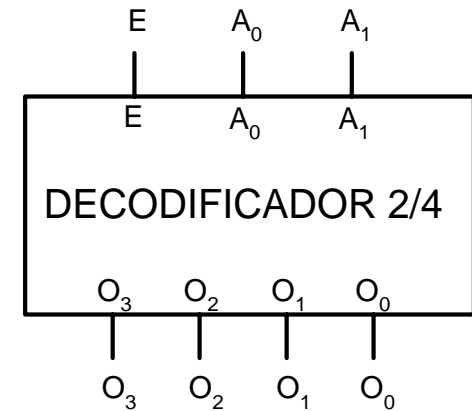
Decodificador de 2 entradas.

Entradas activas a nivel ALTO

Salidas activas a nivel ALTO

$E \rightarrow$ Enable (activa a nivel alto)

Entradas			Salidas			
E	A_1	A_0	O_0	O_1	O_2	O_3
0	X	X	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1



$$O_0 = E \overline{A_1} \overline{A_0} = E m_0$$

$$O_1 = E \overline{A_1} A_0 = E m_1$$

$$O_2 = E A_1 \overline{A_0} = E m_2$$

$$O_3 = E A_1 A_0 = E m_3$$

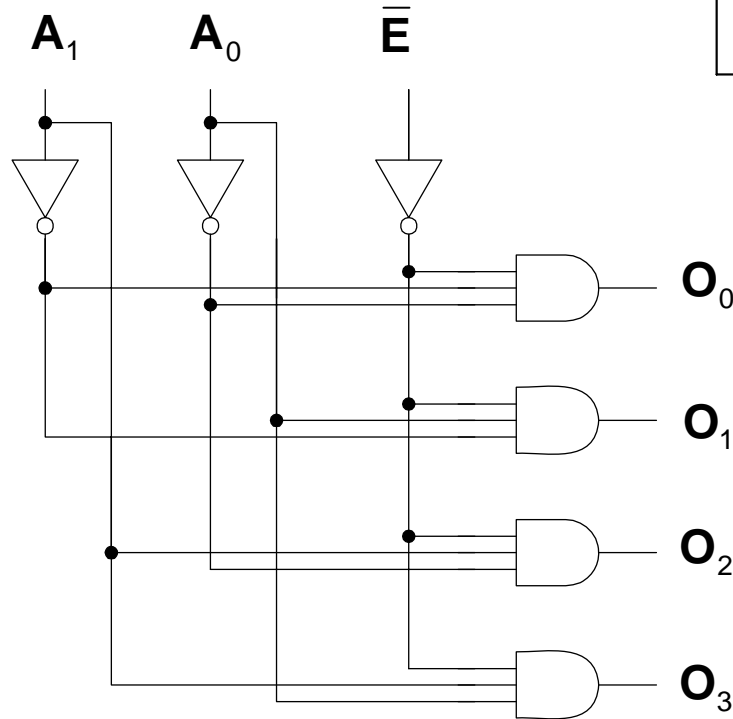
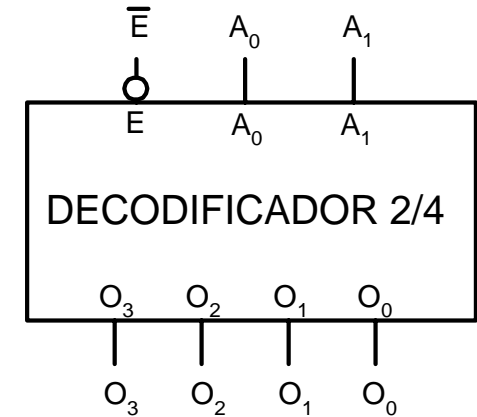
Decodificador de 2 entradas.

Entradas activas a nivel ALTO

Salidas activas a nivel ALTO

$\bar{E} \rightarrow \overline{\text{Enable}}$ (activa a nivel bajo)

Entradas			Salidas			
\bar{E}	A_1	A_0	O_0	O_1	O_2	O_3
1	X	X	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	1	0	0
0	1	0	0	0	1	0
0	1	1	0	0	0	1



$$O_0 = \bar{\bar{E}} \bar{A_1} \bar{A_0} = E m_0$$

$$O_1 = \bar{\bar{E}} \bar{A_1} A_0 = E m_1$$

$$O_2 = \bar{\bar{E}} A_1 \bar{A_0} = E m_2$$

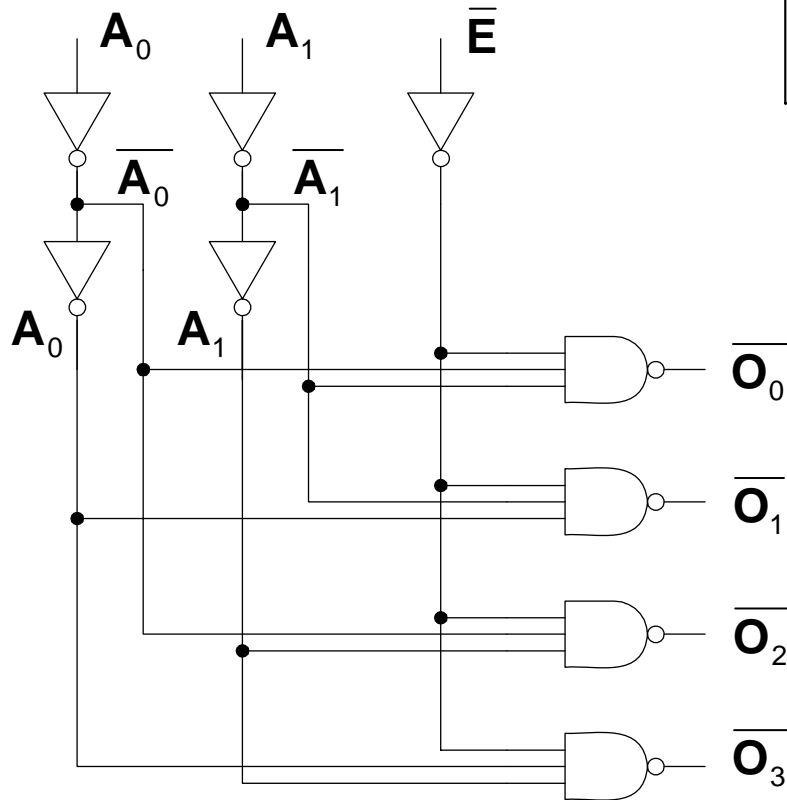
$$O_3 = \bar{\bar{E}} A_1 A_0 = E m_3$$

Decodificador de 2 entradas.

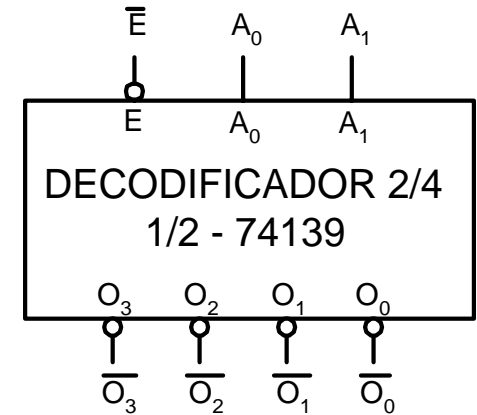
Entradas activas a nivel ALTO

Salidas activas a nivel BAJO

$\bar{E} \rightarrow \overline{\text{Enable}}$ (activa a nivel bajo)



Entradas			Salidas			
\bar{E}	A_1	A_0	\bar{O}_0	\bar{O}_1	\bar{O}_2	\bar{O}_3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0



$$\bar{O}_0 = \bar{E} + A_1 + A_0 = \overline{\bar{E} + A_1 + A_0} = \overline{\bar{E} \bar{A}_1 \bar{A}_0} = \overline{\bar{E} m_0}$$

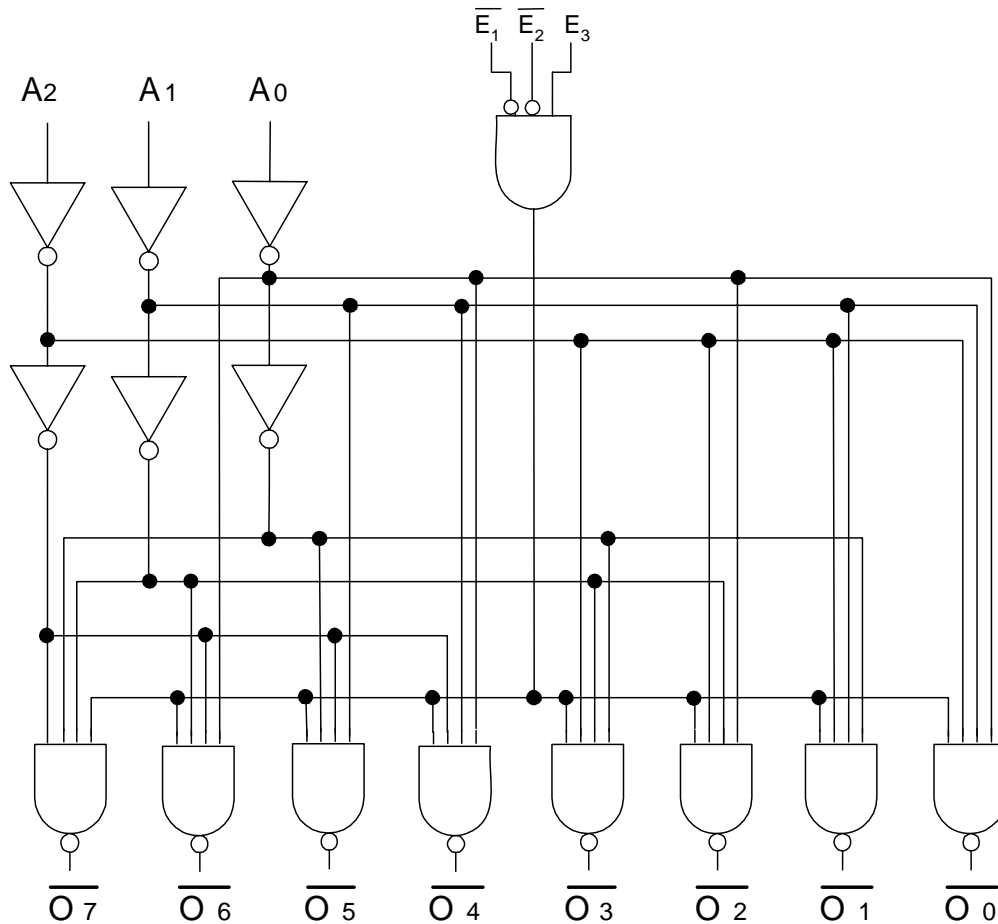
$$\bar{O}_1 = \bar{E} + A_1 + \bar{A}_0 = \overline{\bar{E} + A_1 + \bar{A}_0} = \overline{\bar{E} A_1 A_0} = \overline{\bar{E} m_1}$$

$$\bar{O}_2 = \bar{E} + \bar{A}_1 + A_0 = \overline{\bar{E} + \bar{A}_1 + A_0} = \overline{\bar{E} \bar{A}_1 A_0} = \overline{\bar{E} m_2}$$

$$\bar{O}_3 = \bar{E} + \bar{A}_1 + \bar{A}_0 = \overline{\bar{E} + \bar{A}_1 + \bar{A}_0} = \overline{\bar{E} A_1 A_0} = \overline{\bar{E} m_3}$$

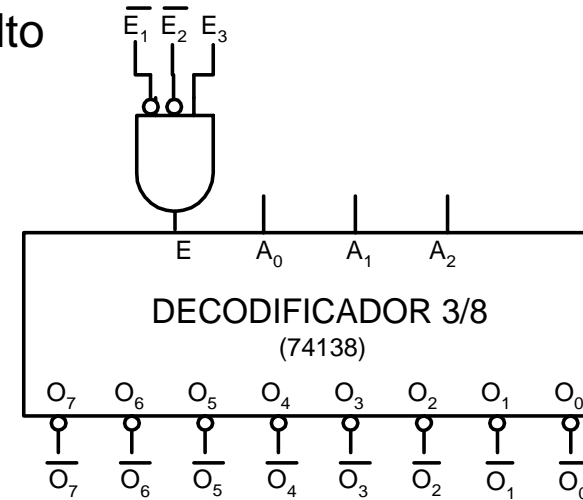
Decodificador Octal.

Decodificador de **3 entradas** activas a nivel alto y **8 salidas** activas a nivel bajo.



$$\overline{O_4} = \overline{E} + \overline{A_2} + \overline{A_1} + \overline{A_0} = \overline{\overline{E} + \overline{A_2} + \overline{A_1} + \overline{A_0}}$$

$$\overline{O_4} = \overline{E A_2 A_1 A_0} = \overline{E m_4}$$



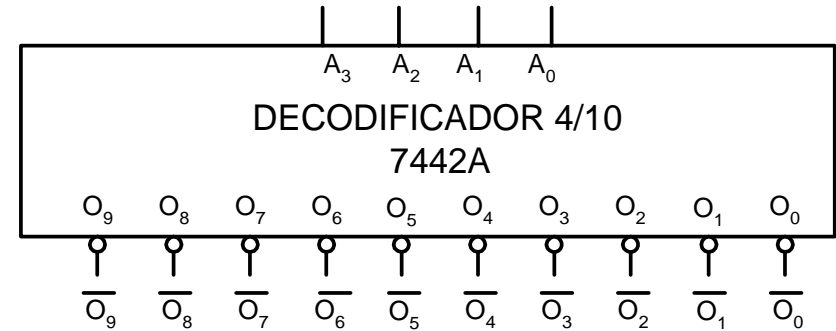
$\overline{E_1}$	$\overline{E_2}$	E_3	E
1	X	X	0
X	1	X	0
X	X	0	0
0	0	1	1

Entradas				Salidas							
E	A_2	A_1	A_0	$\overline{O_0}$	$\overline{O_1}$	$\overline{O_2}$	$\overline{O_3}$	$\overline{O_4}$	$\overline{O_5}$	$\overline{O_6}$	$\overline{O_7}$
0	X	X	X	1	1	1	1	1	1	1	1
1	0	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	0	1	1	1	1	1	1
1	0	1	0	1	1	0	1	1	1	1	1
1	0	1	1	1	1	1	0	1	1	1	1
1	1	0	0	1	1	1	1	0	1	1	1
1	1	0	1	1	1	1	1	1	0	1	1
1	1	1	0	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	1	0

Decodificador BCD - Decimal.

Un decodificador BCD - decimal convierte cada código BCD (8421) en uno de los diez posibles dígitos decimales.

Tiene 4 **entradas** activas a nivel alto y **10 salidas** activas a nivel bajo.

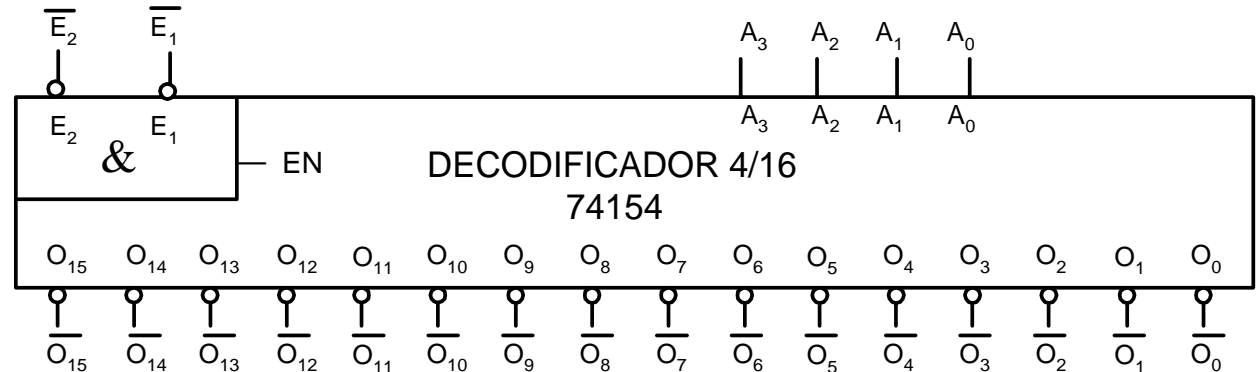


	Entradas				Salidas									
	A ₃	A ₂	A ₁	A ₀	\overline{O}_0	\overline{O}_1	\overline{O}_2	\overline{O}_3	\overline{O}_4	\overline{O}_5	\overline{O}_6	\overline{O}_7	\overline{O}_8	\overline{O}_9
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	0	0	0	1	1	0	1	1	1	1	1	1	1	1
2	0	0	1	0	1	1	0	1	1	1	1	1	1	1
3	0	0	1	1	1	1	1	0	1	1	1	1	1	1
4	0	1	0	0	1	1	1	1	0	1	1	1	1	1
5	0	1	0	1	1	1	1	1	1	0	1	1	1	1
6	0	1	1	0	1	1	1	1	1	1	0	1	1	1
7	0	1	1	1	1	1	1	1	1	1	1	0	1	1
8	1	0	0	0	1	1	1	1	1	1	1	1	0	1
9	1	0	0	1	1	1	1	1	1	1	1	1	1	0
10	1	0	1	0	1	1	1	1	1	1	1	1	1	1
...									
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Decodificador Hexadecimal.

Decodificador hexadecimal
(4 entradas activas a nivel alto -
16 salidas activas a nivel bajo).

Habilitado $\Rightarrow \overline{E}_1 = \overline{E}_2 = 0$



Entradas				Salidas															
A_3	A_2	A_1	A_0	\overline{O}_0	\overline{O}_1	\overline{O}_2	\overline{O}_3	\overline{O}_4	\overline{O}_5	\overline{O}_6	\overline{O}_7	\overline{O}_8	\overline{O}_9	\overline{O}_{10}	\overline{O}_{11}	\overline{O}_{12}	\overline{O}_{13}	\overline{O}_{14}	\overline{O}_{15}
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
0	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1

Implementación de funciones lógicas con decodificadores.

Ejemplo: Obtener la función **F** a partir de un decodificador **BCD-decimal 7442A**.

	c	b	a	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

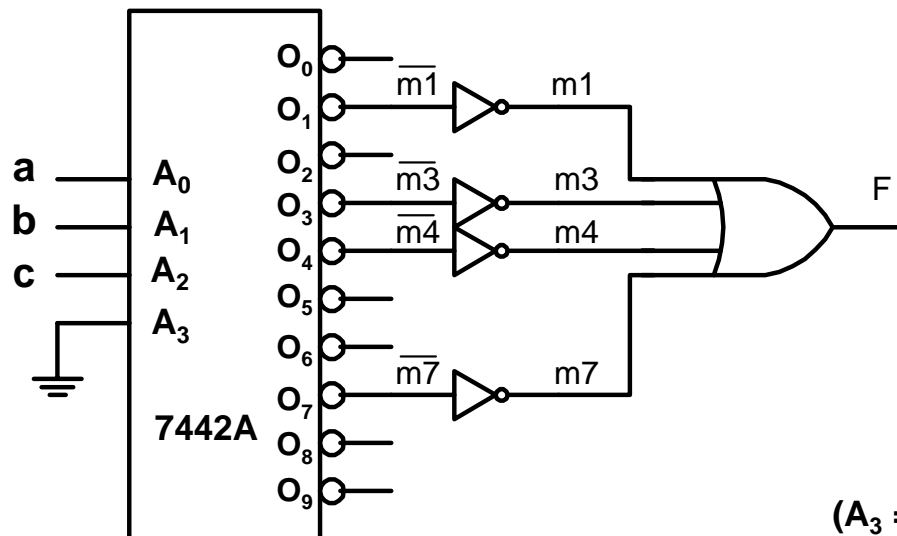
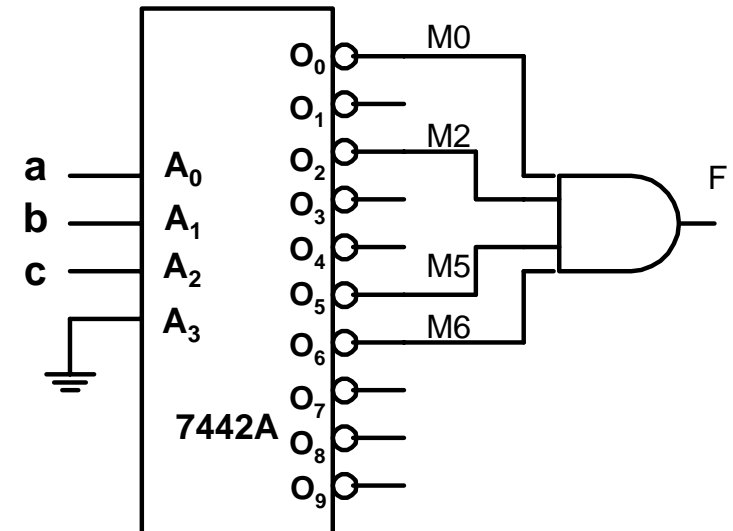
$$F = \sum_3 (1, 3, 4, 7) = m_1 + m_3 + m_4 + m_7$$

$$F = \bar{c}\bar{b}a + \bar{c}ba + c\bar{b}\bar{a} + cba$$

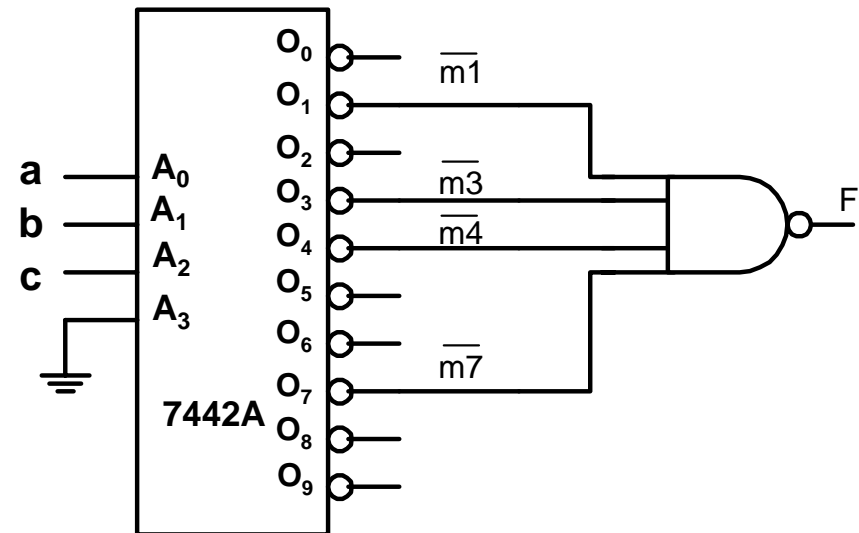
$$F = \overline{m_1 + m_3 + m_4 + m_7} = \overline{m_1 \cdot m_3 \cdot m_4 \cdot m_7}$$

$$F = \prod_3 (0, 2, 5, 6)$$

$$F = M_0 \cdot M_2 \cdot M_5 \cdot M_6 = \bar{m}_0 \cdot \bar{m}_2 \cdot \bar{m}_5 \cdot \bar{m}_6$$



(A₃ = MSB)

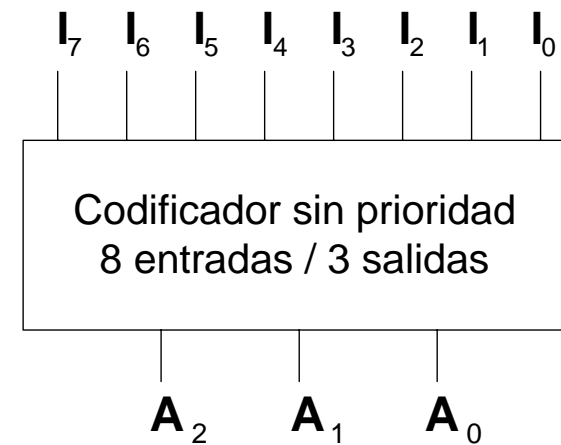


Codificadores.

- Son circuitos combinacionales de **2^N entradas** y **N salidas**.
- Su función básica es codificar en binario la información que aparece en sus salidas.

Codificador sin prioridad.

Entradas								Salidas		
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	A_2	A_1	A_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0



Funciona bien si sólo hay una entrada activa.

Si hay dos entrada activas, por ejemplo $I_4 = I_3 = 1$

$$\begin{array}{rcl}
 I_3 = 1 & \Rightarrow & \begin{array}{ccc} A_2 & A_1 & A_0 \\ 0 & 1 & 1 \end{array} \\
 I_4 = 1 & \Rightarrow & \begin{array}{ccc} A_2 & A_1 & A_0 \\ 1 & 0 & 0 \end{array} \\
 \text{(OR)} & & \begin{array}{ccc} 1 & 1 & 1 \end{array} = I_7 \text{ ERROR}
 \end{array}$$

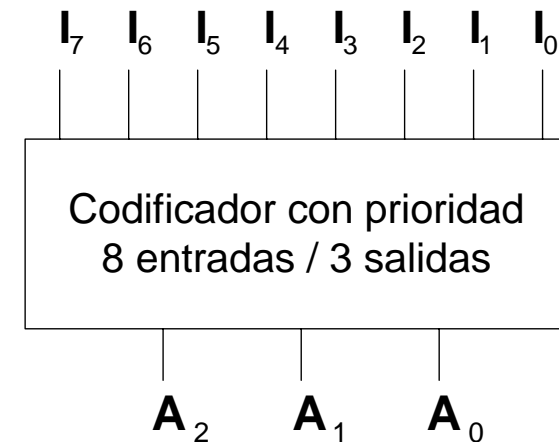
Para solventar este error sólo debe activarse una entrada o utilizar CODIFICADORES CON PRIORIDAD.

Codificador con prioridad.

Codifican la salida correspondiente a la entrada de mayor peso.

Codificador con prioridad (entradas y salidas activas a nivel alto).

Entradas								Salidas		
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	A_2	A_1	A_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	X	0	0	1
0	0	0	0	0	1	X	X	0	1	0
0	0	0	0	1	X	X	X	0	1	1
0	0	0	1	X	X	X	X	1	0	0
0	0	1	X	X	X	X	X	1	0	1
0	1	X	X	X	X	X	X	1	1	0
1	X	X	X	X	X	X	X	1	1	1
0	0	0	0	0	0	0	0	0	0	0



Si hay dos entrada activas, por ejemplo $I_4 = I_3 = 1$

$$\begin{array}{rcl} I_3 = 1 & \Rightarrow & \begin{array}{ccc} A_2 & A_1 & A_0 \\ 0 & 1 & 1 \end{array} \\ I_4 = 1 & \Rightarrow & \begin{array}{ccc} 1 & 0 & 0 \end{array} \\ \hline & & 1 \quad 0 \quad 0 = I_4 \end{array}$$

Codificador con prioridad

(Entradas activas a nivel alto - salidas activas a nivel bajo).

Entradas								Salidas		
I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	$\overline{A_2}$	$\overline{A_1}$	$\overline{A_0}$
0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	1	X	1	1	0
0	0	0	0	0	1	X	X	1	0	1
0	0	0	0	1	X	X	X	1	0	0
0	0	0	1	X	X	X	X	0	1	1
0	0	1	X	X	X	X	X	0	1	0
0	1	X	X	X	X	X	X	0	0	1
1	X	X	X	X	X	X	X	0	0	0
0	0	0	0	0	0	0	0	1	1	1



Codificador con prioridad

(Entradas activas a nivel bajo - salidas activas a nivel bajo).

Entradas								Salidas		
$\overline{I_7}$	$\overline{I_6}$	$\overline{I_5}$	$\overline{I_4}$	$\overline{I_3}$	$\overline{I_2}$	$\overline{I_1}$	$\overline{I_0}$	$\overline{A_2}$	$\overline{A_1}$	$\overline{A_0}$
1	1	1	1	1	1	1	0	1	1	1
1	1	1	1	1	1	0	X	1	1	0
1	1	1	1	1	0	X	X	1	0	1
1	1	1	1	0	X	X	X	1	0	0
1	1	1	0	X	X	X	X	0	1	1
1	1	0	X	X	X	X	X	0	1	0
1	0	X	X	X	X	X	X	0	0	1
0	X	X	X	X	X	X	X	0	0	0
1	1	1	1	1	1	1	1	1	1	1



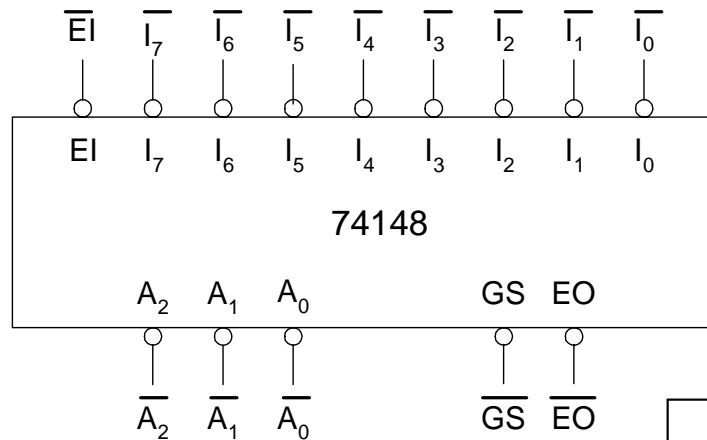
Ejemplo: Entrada = 1 0 1 1 0 1 0 1

$\overline{I_6}$ $6_{(2)} = 110 \Rightarrow$ Salida 0 0 1

Entrada = 1 1 1 1 0 1 0 1

$\overline{I_3}$ $3_{(2)} = 011 \Rightarrow$ Salida 1 0 0

Codificador de prioridad octal - binario (74148).

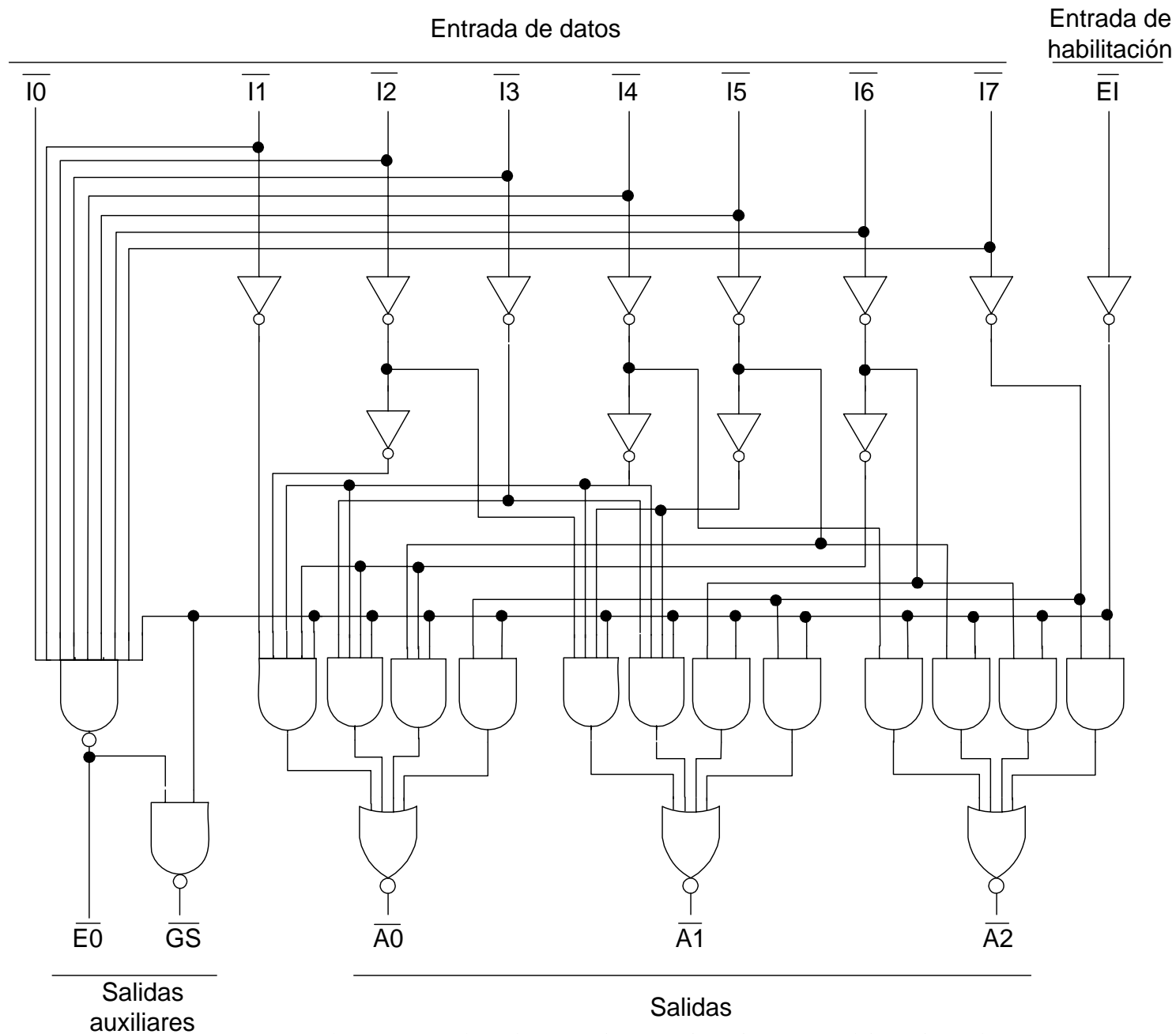


$\overline{\text{GS}}$ → Señal de grupo

$\overline{\text{EO}}$ → Salida de habilitación

$\overline{\text{EI}}$	Entradas								Salidas			Aux.	
	$\overline{\text{I}}_7$	$\overline{\text{I}}_6$	$\overline{\text{I}}_5$	$\overline{\text{I}}_4$	$\overline{\text{I}}_3$	$\overline{\text{I}}_2$	$\overline{\text{I}}_1$	$\overline{\text{I}}_0$	$\overline{\text{A}}_2$	$\overline{\text{A}}_1$	$\overline{\text{A}}_0$	$\overline{\text{GS}}$	$\overline{\text{EO}}$
1	X	X	X	X	X	X	X	X	1	1	1	1	1
0	1	1	1	1	1	1	1	0	1	1	1	0	1
0	1	1	1	1	1	1	0	X	1	1	0	0	1
0	1	1	1	1	0	X	X	X	1	0	0	0	1
0	1	1	1	0	X	X	X	X	0	1	1	0	1
0	1	1	0	X	X	X	X	X	0	1	0	0	1
0	1	0	X	X	X	X	X	X	0	0	1	0	1
0	0	X	X	X	X	X	X	X	0	0	0	0	1
0	1	1	1	1	1	1	1	1	1	1	1	1	0

Codificador de prioridad octal - binario (74148).



Multiplexores.

Son circuitos combinacionales de **2^N entradas de datos**, **N entradas de selección** y **1 salida**.

Su función básica es transmitir los datos desde la entrada seleccionada a la salida que es única.

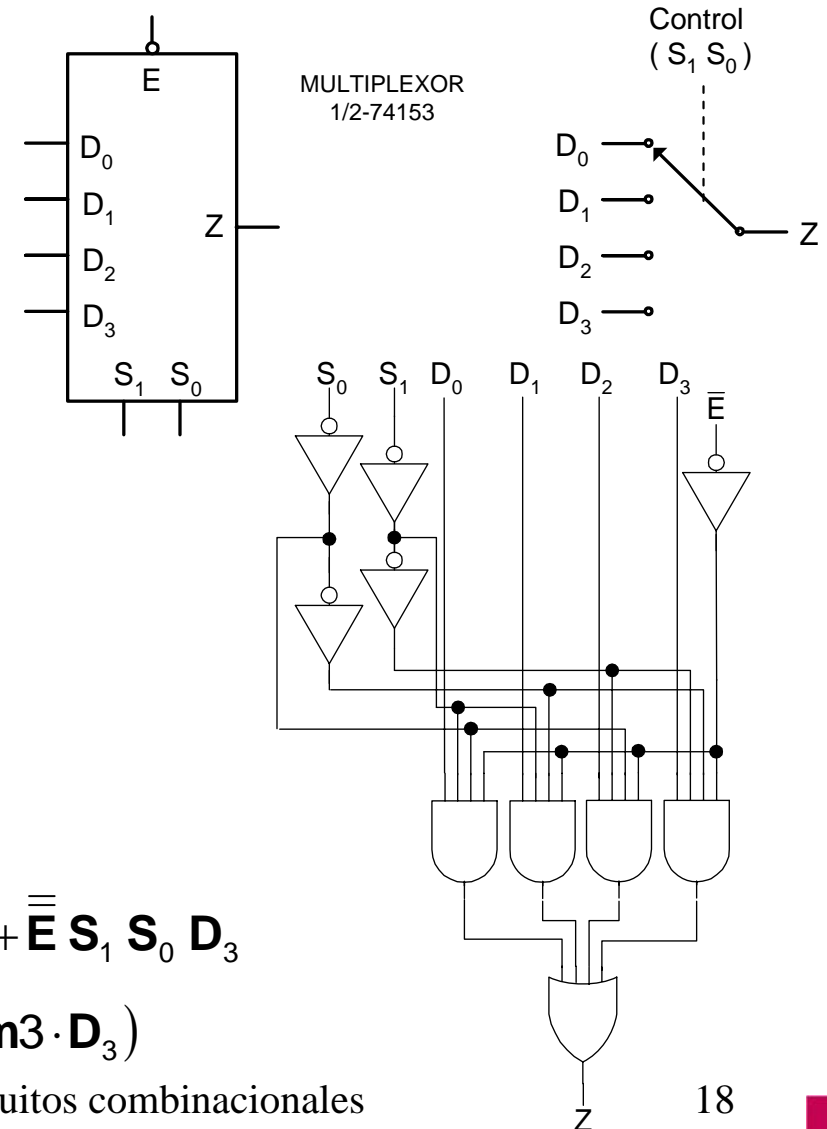
Multiplexor de 4 entradas.

Enable	Entradas Selección		Entradas de Datos				Salida	
\bar{E}	S_1	S_0	D_0	D_1	D_2	D_3	Z	
1	X	X	X	X	X	X	0	0
0	0	0	0	X	X	X	0	D_0
0	0	0	1	X	X	X	1	
0	0	1	X	0	X	X	0	D_1
0	0	1	X	1	X	X	1	
0	1	0	X	X	0	X	0	D_2
0	1	0	X	X	1	X	1	
0	1	1	X	X	X	0	0	D_3
0	1	1	X	X	X	1	1	

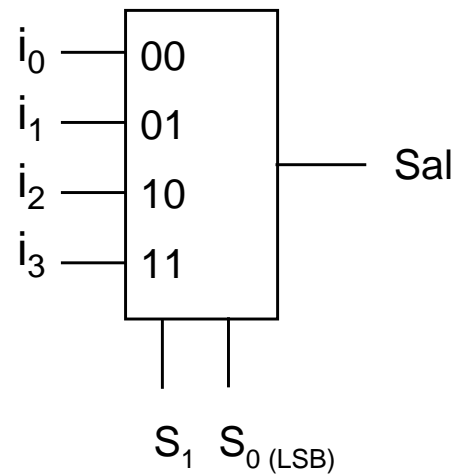
$$Z = \bar{E} \bar{S}_1 \bar{S}_0 D_0 + \bar{E} \bar{S}_1 S_0 D_1 + \bar{E} S_1 \bar{S}_0 D_2 + \bar{E} S_1 S_0 D_3$$

$$Z = E (m_0 \cdot D_0 + m_1 \cdot D_1 + m_2 \cdot D_2 + m_3 \cdot D_3)$$

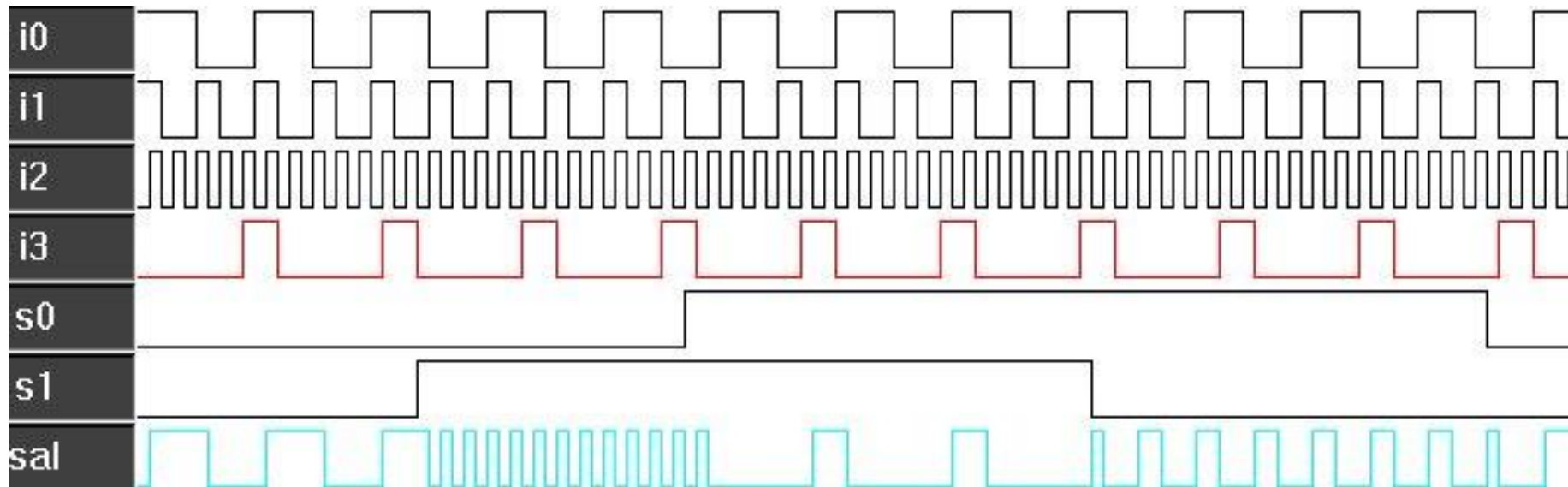
Fundamentos de Electrónica - Circuitos combinacionales



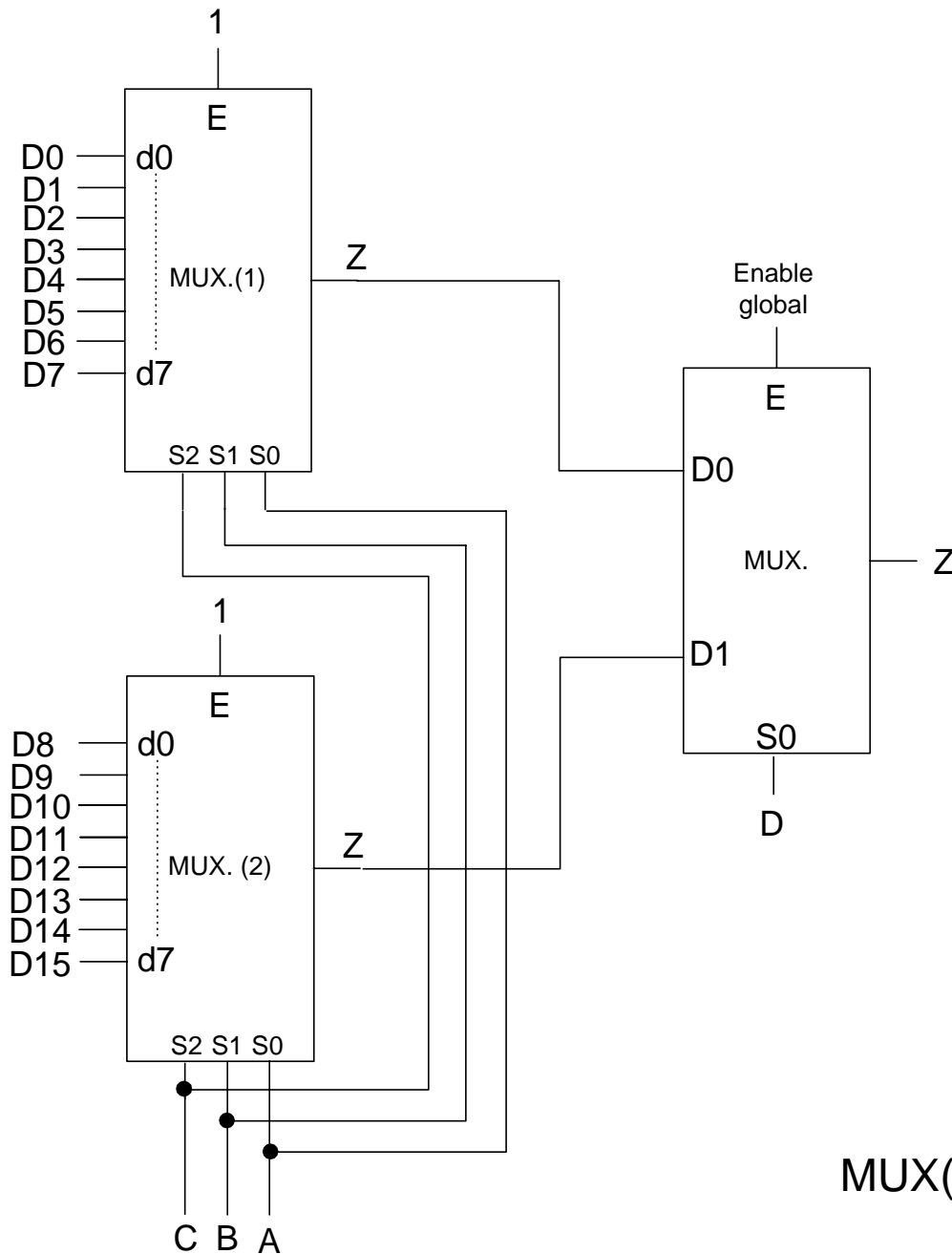
Multiplexores.



$$Sal = \overline{S_1} \overline{S_0} i_0 + \overline{S_1} S_0 i_1 + S_1 \overline{S_0} i_2 + S_1 S_0 i_3$$



Conexión de multiplexores.



	D	C	B	A	
D0	0	0	0	0	D = 0
D1	0	0	0	1	
D2	0	0	1	0	
D3	0	0	1	1	
D4	0	1	0	0	
D5	0	1	0	1	
D6	0	1	1	0	
D7	0	1	1	1	
D8	1	0	0	0	D = 1
D9	1	0	0	1	
D10	1	0	1	0	
D11	1	0	1	1	
D12	1	1	0	0	
D13	1	1	0	1	
D14	1	1	1	0	
D15	1	1	1	1	

MUX(1) y MUX(2) → S2 S1 S0

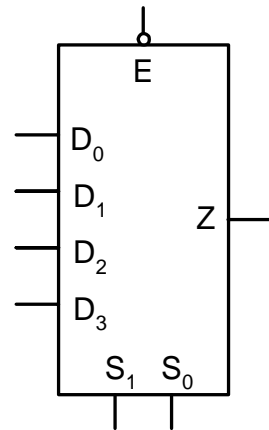
Funciones lógicas con multiplexores.

Ejemplo 1: Mediante un multiplexor de 4 entradas, implementar la función: $f = \Sigma_3(0, 1, 4, 7)$

$S_1 \ S_0$			Z
A	B	C	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

$$f = \underbrace{\bar{A}\bar{B}\bar{C}} + \underbrace{\bar{A}\bar{B}C} + \underbrace{A\bar{B}\bar{C}} + ABC = \bar{B}\bar{C} + \bar{A}\bar{B}C + ABC$$

$$\bar{B}\bar{C}(\bar{A} + A) = \bar{B}\bar{C}$$



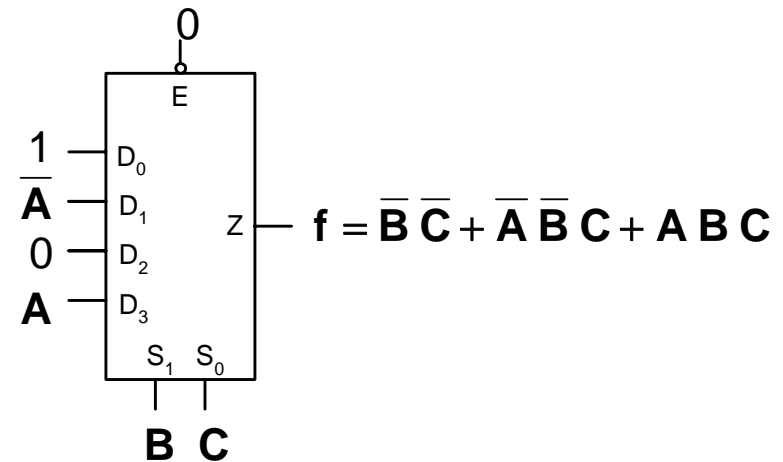
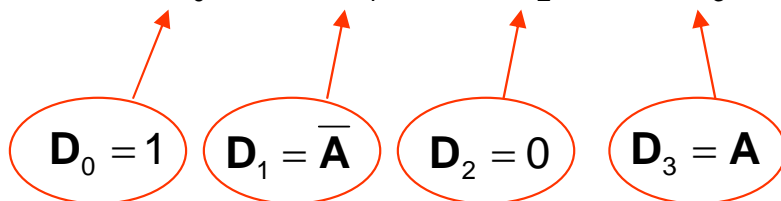
S_1	S_0	
0	0	m0
0	1	m1
1	0	m2
1	1	m3

$$Z = m0 D_0 + m1 D_1 + m2 D_2 + m3 D_3$$

$$Z = f \rightarrow S_1 = B \text{ y } S_0 = C$$

D_0, D_1, D_2 y D_3 pueden ser: $\bar{A}, A, 0, 1$

$$Z = \bar{B}\bar{C}D_0 + \bar{B}CD_1 + B\bar{C}D_2 + BCD_3$$



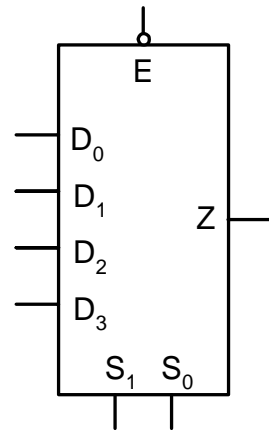
Funciones lógicas con multiplexores.

Ejemplo 1: Mediante un multiplexor de 4 entradas, implementar la función: $f = \Sigma_3(0, 1, 4, 7)$

$S_1 \ S_0$		Z	
A	B	C	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

$$f = \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + A\overline{B}\overline{C} + ABC = \overline{A}\overline{B} + A\overline{B}\overline{C} + ABC$$

$$\overline{A}\overline{B}(\overline{C} + C) = \overline{A}\overline{B}$$



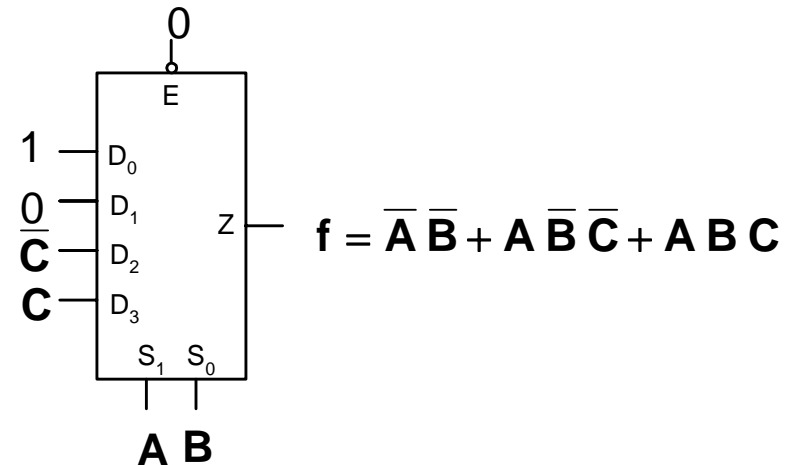
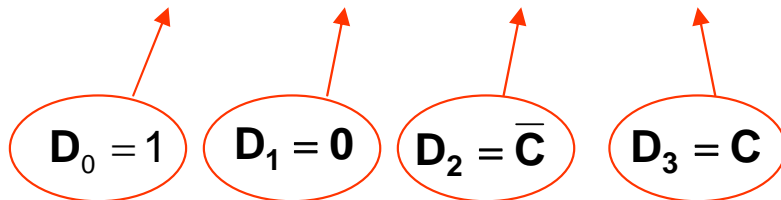
S_1	S_0	
0	0	m0
0	1	m1
1	0	m2
1	1	m3

$$Z = m0 D_0 + m1 D_1 + m2 D_2 + m3 D_3$$

$$Z = f \rightarrow S_1 = A \text{ y } S_0 = B$$

D_0, D_1, D_2 y D_3 pueden ser: $\overline{C}, C, 0, 1$

$$Z = \overline{A}\overline{B}D_0 + \overline{A}BD_1 + A\overline{B}D_2 + ABD_3$$



Demultiplexores.

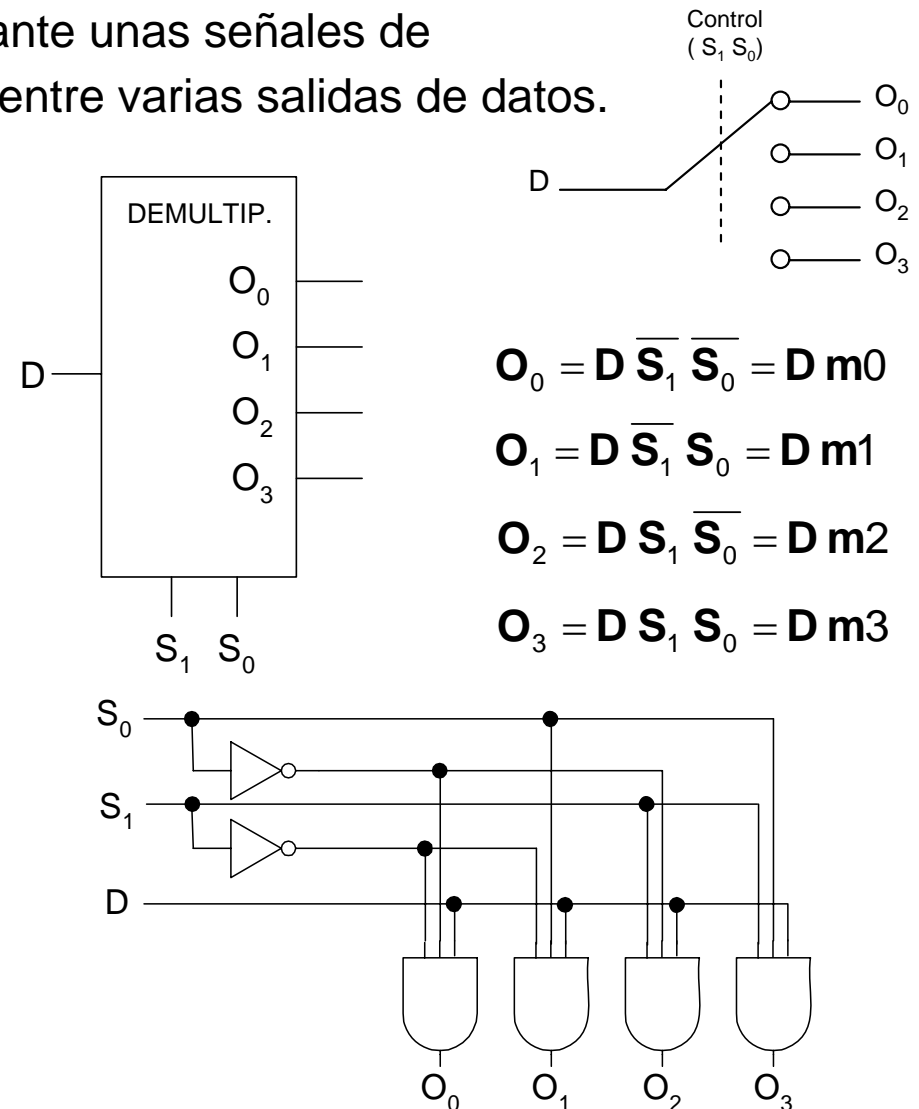
Son circuitos combinacionales de **1 entrada**, **N entradas de selección** y **2^N salidas**.

Realizan la función contraria a la de los multiplexores.

Tienen una única entrada de datos que, mediante unas señales de control, se pone en comunicación con una de entre varias salidas de datos.

Demultiplexor de 4 salidas.

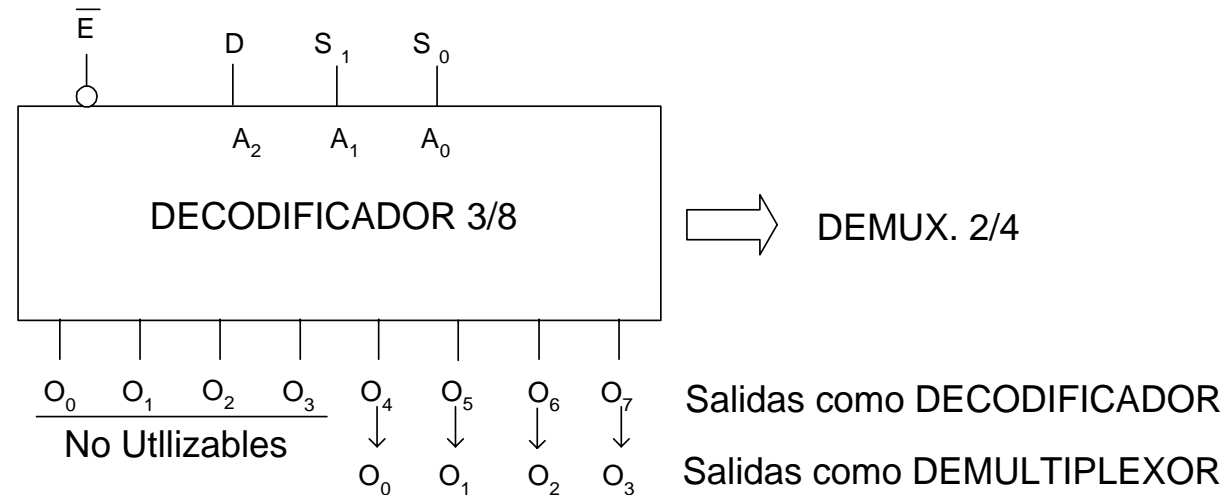
Entradas Selección		Entrada de Datos	Salidas			
S_1	S_0	D	O_0	O_1	O_2	O_3
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	0	1	0
1	1	0	0	0	0	0
1	1	1	0	0	0	1



Demultiplexor a partir de un decodificador.

A partir de un decodificador 1 de 8 se puede construir un demultiplexor.

$\overline{E} = 0 \Rightarrow$ Habilitado



A_2	A_1	A_0	O_0	O_1	O_2	O_3	O_4	O_5	O_6	O_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1
D	S_1	S_0					O_0	O_1	O_2	O_3

\Rightarrow **DECODIFICADOR**

$$O_0 = D \overline{S_1} \overline{S_0} = D m_0$$

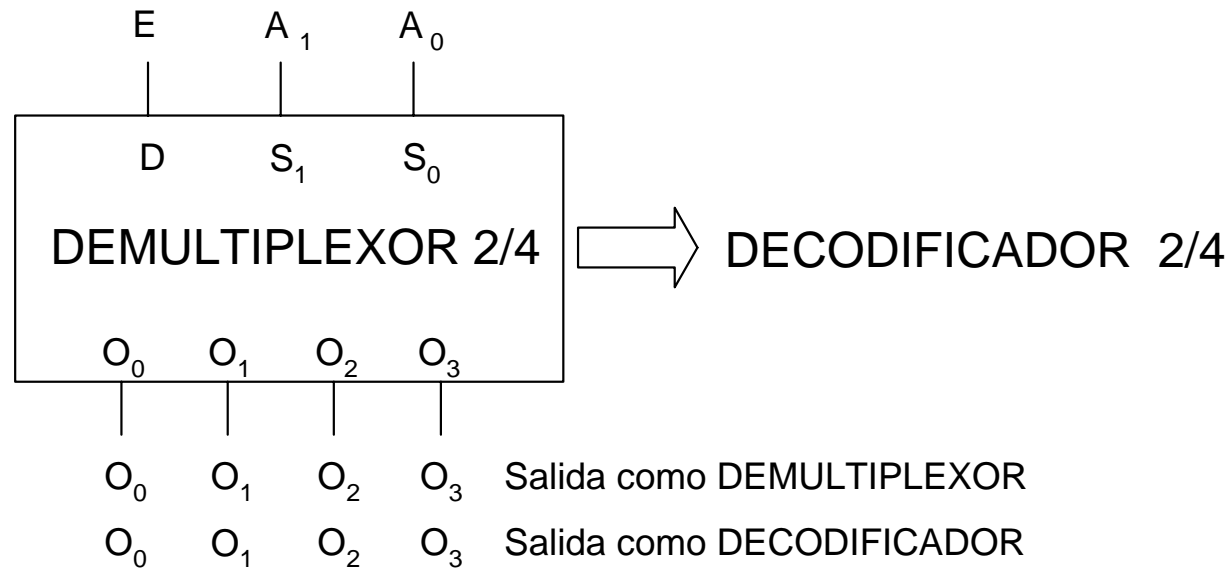
$$O_1 = D \overline{S_1} S_0 = D m_1$$

$$O_2 = D S_1 \overline{S_0} = D m_2$$

$$O_3 = D S_1 S_0 = D m_3$$

\Rightarrow **DEMULTIPLEXOR**

Demultiplexor funcionando como decodificador.



$E = 1 \Rightarrow$ Habilitado

D	S_1	S_0	O_0	O_1	O_2	O_3
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
E	A_1	A_0	O_0	O_1	O_2	O_3

\Rightarrow DEMULTIPLEXOR

$$O_0 = D \overline{S_1} \overline{S_0} = E \overline{A_1} \overline{A_0}$$

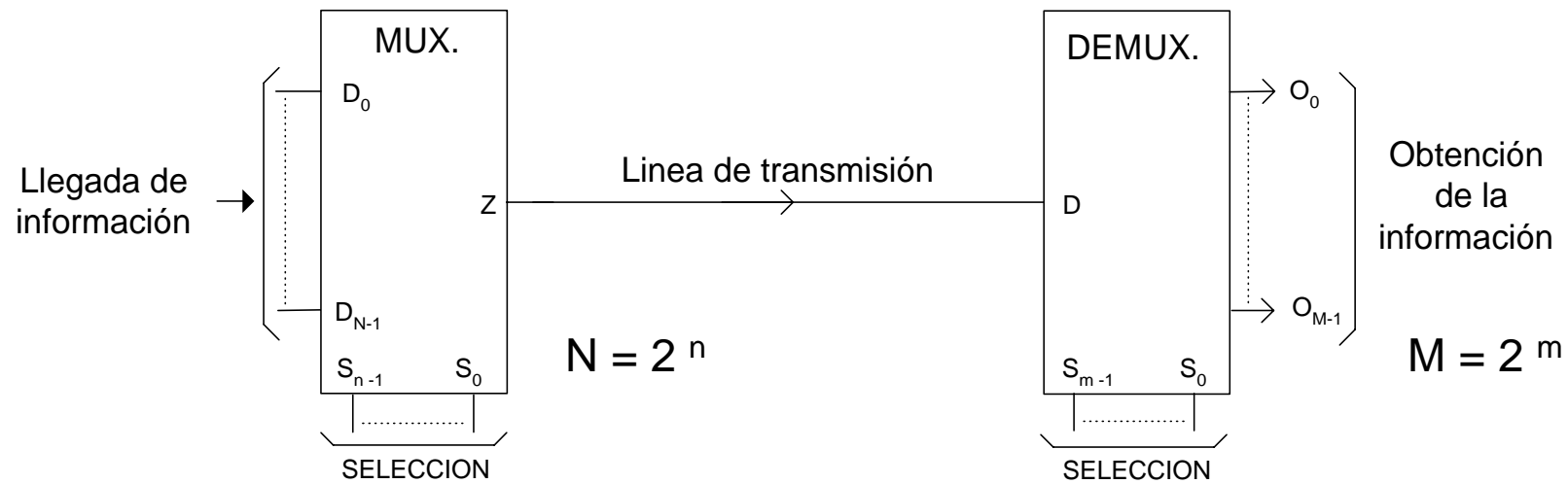
$$O_1 = D \overline{S_1} S_0 = E \overline{A_1} A_0$$

$$O_2 = D S_1 \overline{S_0} = E A_1 \overline{A_0}$$

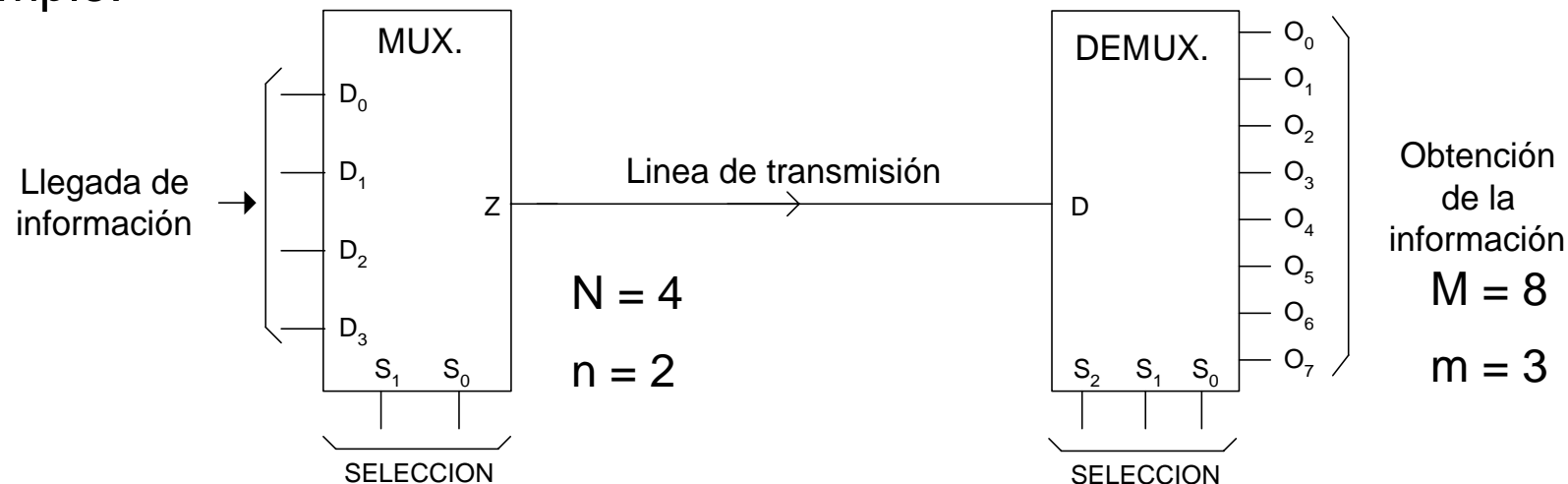
$$O_3 = D S_1 S_0 = E A_1 A_0$$

\Rightarrow DECODIFICADOR

Utilización de multiplexores y demultiplexores en la transmisión de datos por una línea.

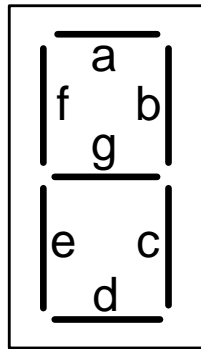
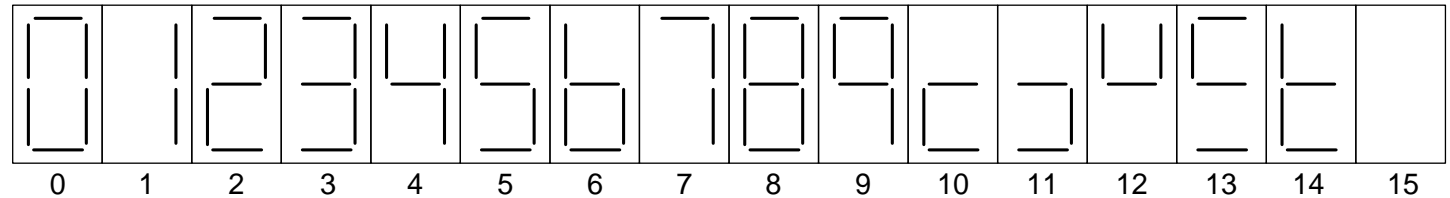
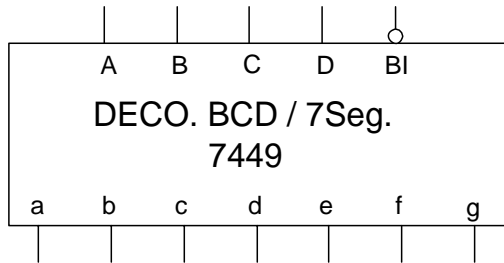


Ejemplo:



Convertidores de código.

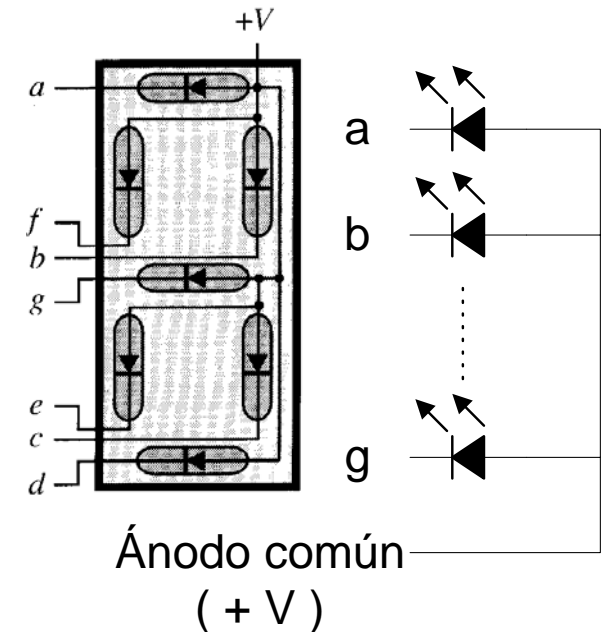
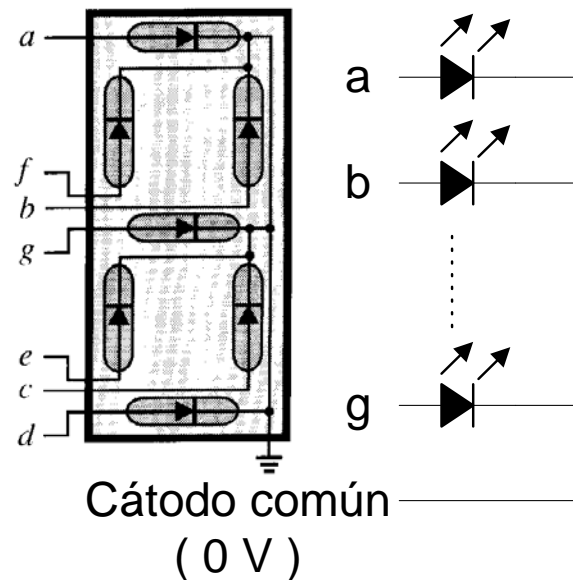
Decodificador BCD a 7 segmentos.



CONVERTIDOR BCD a 7 SEGMENTOS: Circuito combinacional con el que, al introducir un código BCD en la entrada, se obtiene un código de 7 segmentos que permite activar un *display* donde aparece en decimal el valor del código BCD introducido.

Display de 7 segmentos:

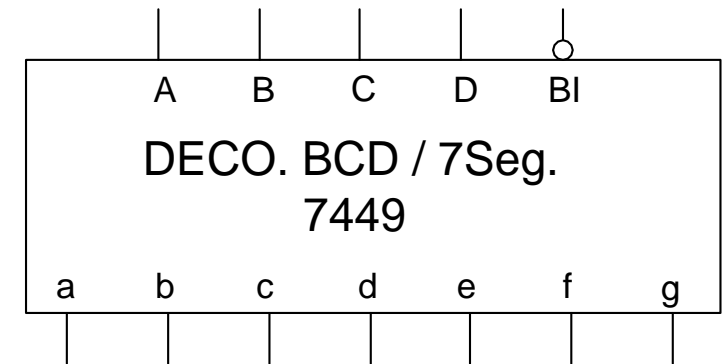
Son 7 diodos LED que pueden tener sus cátodos o sus ánodos comunes. Los diodos LED se iluminan al estar polarizados en directo.



Decodificador BCD a 7 segmentos (7449).

Es un decodificador excitador para *display* de cátodo común salidas del decodificador activas a nivel alto.

Nº	$\overline{\text{BI}}$	D	C	B	A	a	b	c	d	e	f	g
0	1	0	0	0	0	1	1	1	1	1	1	0
1	1	0	0	0	1	0	1	1	0	0	0	0
2	1	0	0	1	0	1	1	0	1	1	0	1
3	1	0	0	1	1	1	1	1	1	0	0	1
4	1	0	1	0	0	0	1	1	0	0	1	1
5	1	0	1	0	1	1	0	1	1	0	1	1
6	1	0	1	1	0	0	0	1	1	1	1	1
7	1	0	1	1	1	1	1	1	0	0	0	0
8	1	1	0	0	0	1	1	1	1	1	1	1
9	1	1	0	0	1	1	1	1	0	0	1	1
10	1	1	0	1	0	0	0	0	1	1	0	1
11	1	1	0	1	1	0	0	1	1	0	0	1
12	1	1	1	0	0	0	1	0	0	0	1	1
13	1	1	1	0	1	1	0	0	1	0	1	1
14	1	1	1	1	0	0	0	0	1	1	1	1
15	1	1	1	1	1	0	0	0	0	0	0	0
	0	X	X	X	X	0	0	0	0	0	0	0

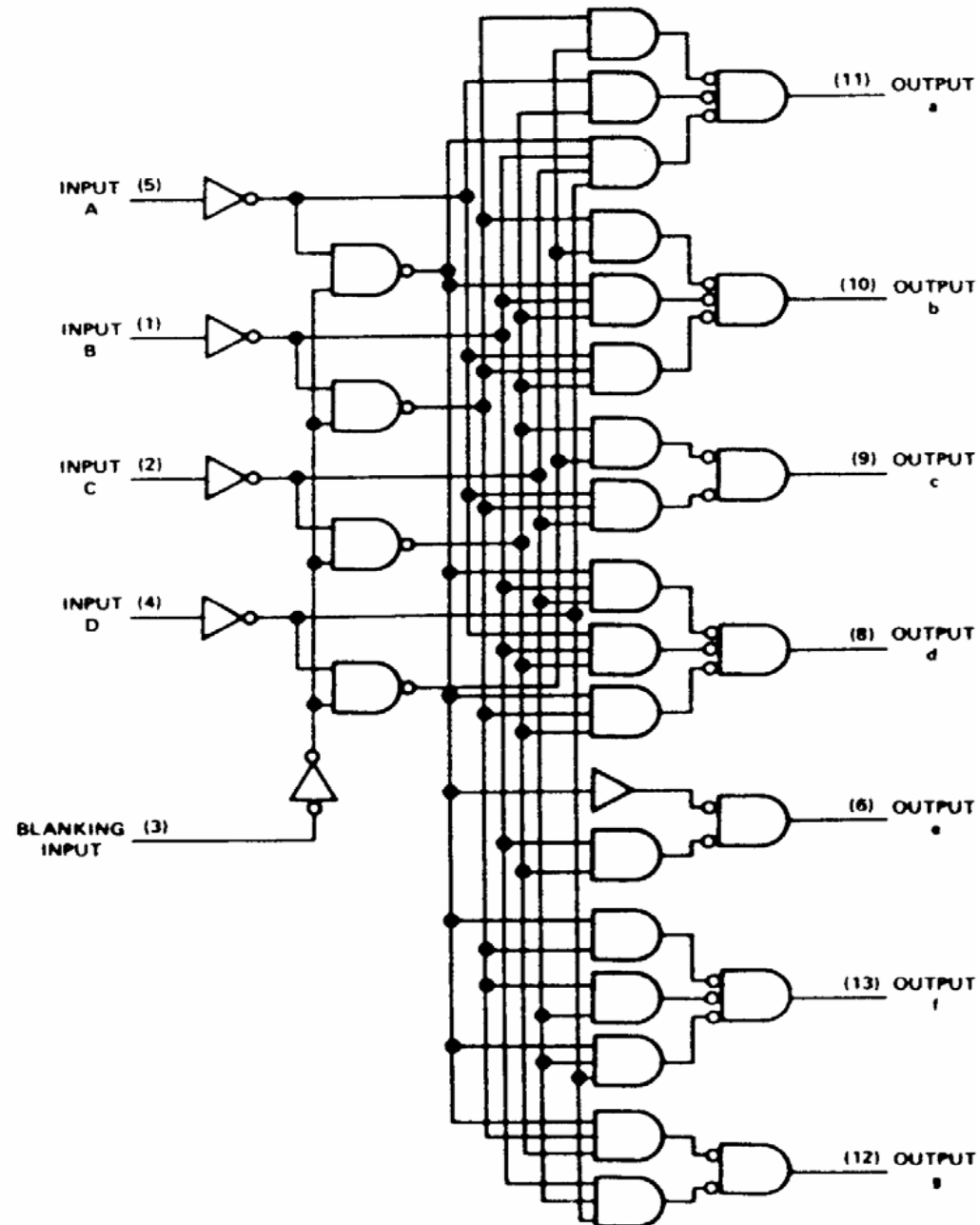


$\overline{\text{BI}} \rightarrow$ BLANKING INPUT

$$\overline{a} = \overline{D} \overline{C} \overline{B} \overline{A} + \overline{A} \overline{C} + \overline{B} \overline{D}$$

$$a = \overline{\overline{D} \overline{C} \overline{B} \overline{A}} \cdot \overline{\overline{A} \overline{C}} \cdot \overline{\overline{B} \overline{D}}$$

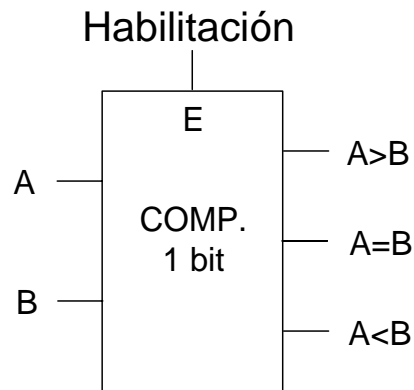
Decodificador BCD a 7 segmentos (7449).



Comparadores.

Son circuitos combinacionales que comparan bit a bit dos números con la misma cantidad de bits.

Comparador de 1 bit.

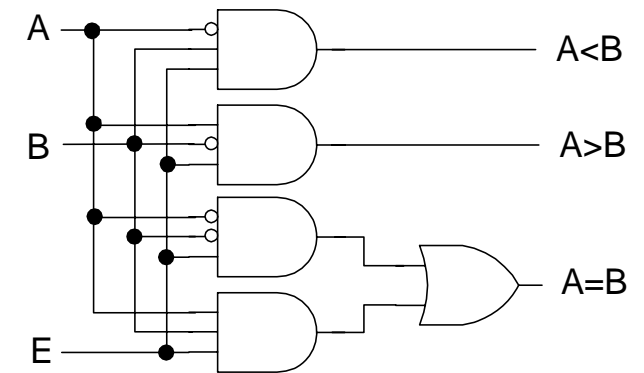


E	A	B	A > B	A = B	A < B
0	X	X	0	0	0
1	0	0	0	1	0
1	0	1	0	0	1
1	1	0	1	0	0
1	1	1	0	1	0

$$(A > B) = E A \bar{B}$$

$$(A = B) = E (\bar{A} \bar{B} + A B) = E (\overline{A \oplus B})$$

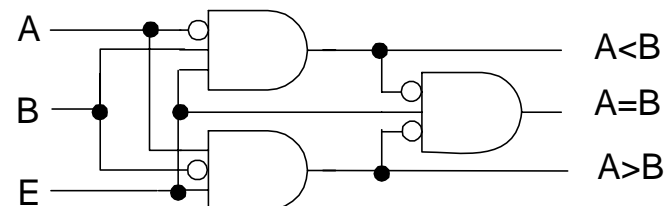
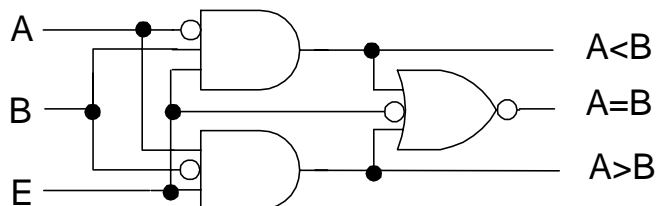
$$(A < B) = E \bar{A} B$$



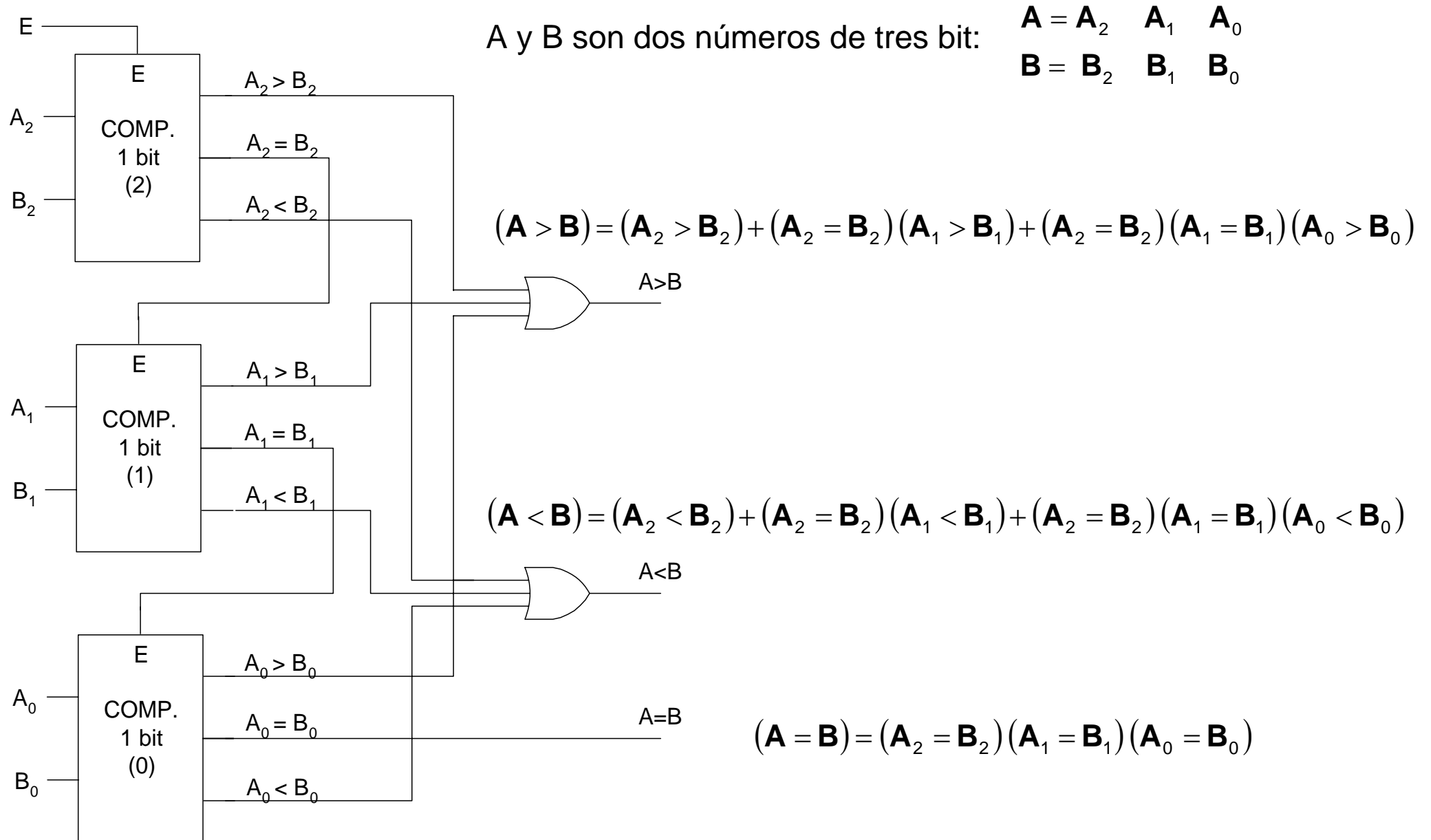
Otra forma: $(A > B) = E A \bar{B}$ $(A < B) = E \bar{A} B$

$$(A \neq B) = (A > B) + (A < B) + \bar{E}$$

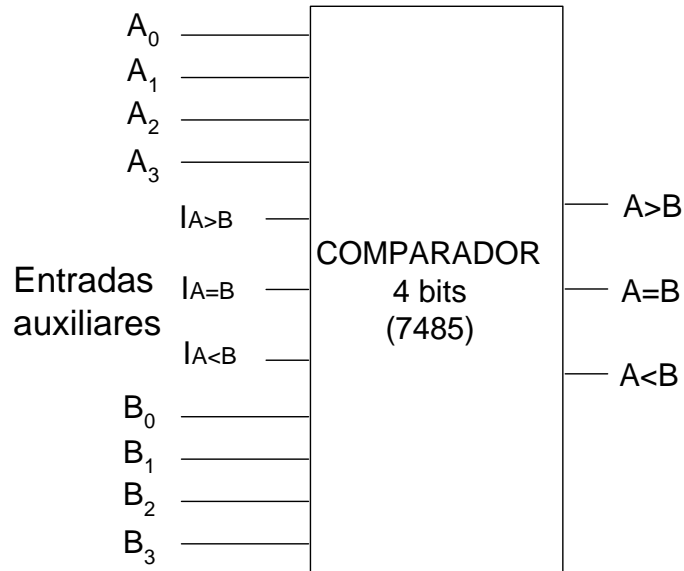
$$(A = B) = \overline{(A \neq B)} = \overline{(A > B) + (A < B) + \bar{E}} = \overline{(A > B)} \overline{(A < B)} E$$



Comparador de 3 bits.



Comparador de 4 bits (7485).



A = **A**₃ **A**₂ **A**₁ **A**₀

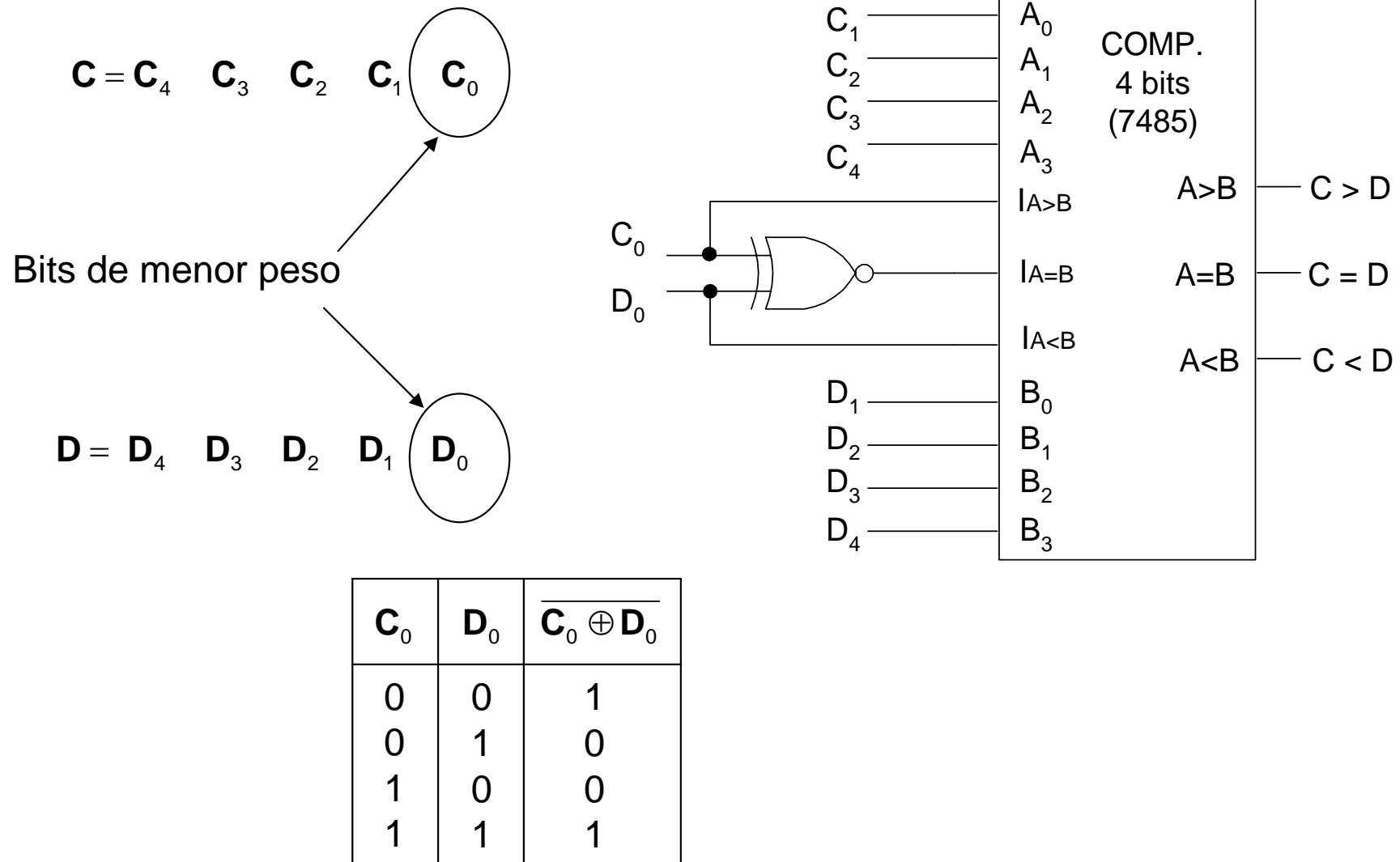
B = **B**₃ **B**₂ **B**₁ **B**₀

Entradas de comparación				Ent. Aux.			Salidas		
(A ₃ , B ₃)	(A ₂ , B ₂)	(A ₁ , B ₁)	(A ₀ , B ₀)	I _{A>B}	I _{A<B}	I _{A=B}	(A > B)	(A < B)	(A = B)
A ₃ > B ₃	X	X	X	X	X	X	1	0	0
A ₃ = B ₃	A ₂ > B ₂	X	X	X	X	X	1	0	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ > B ₁	X	X	X	X	1	0	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ = B ₁	A ₀ > B ₀	X	X	X	1	0	0
A ₃ < B ₃	X	X	X	X	X	X	0	1	0
A ₃ = B ₃	A ₂ < B ₂	X	X	X	X	X	0	1	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ < B ₁	X	X	X	X	0	1	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ = B ₁	A ₀ < B ₀	X	X	X	0	1	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ = B ₁	A ₀ = B ₀	1	0	0	1	0	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ = B ₁	A ₀ = B ₀	0	1	0	0	1	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ = B ₁	A ₀ = B ₀	X	X	1	0	0	1
A ₃ = B ₃	A ₂ = B ₂	A ₁ = B ₁	A ₀ = B ₀	1	1	0	0	0	0
A ₃ = B ₃	A ₂ = B ₂	A ₁ = B ₁	A ₀ = B ₀	0	0	0	1	1	0

Las entradas auxiliares sirven para la conexión en cascada de varios comparadores.

En un comparador de 4 bit estas entradas deben ser: $I_{A>B} = I_{A<B} = 0$; $I_{A=B} = 1$.

Comparador de dos números de 5 bits.



Comparador de dos números de 8 bits.

