



# Tema 3

Tema 3: Lógica  
Combinacional (II): Ruta de  
Datos.

### **TEMA 3: LÓGICA COMBINACIONAL (II): RUTA DE DATOS**

- *Contexto*
- *Conocimiento Previo Necesario*
- *Objetivos del Tema*
- *Guía de Estudio*
- *Contenido del Tema*
  - 3.1. Circuitos Selectores de Datos (Multiplexos)
  - 3.2. Demultiplexos
  - 3.3. Codificadores con Prioridad
  - 3.4. Amplificadores (buffers-drivers) y Transmisores-receptores de Bus
  - 3.5. Problemas
- *Preparación de la Evaluación*

### +++ OBJETIVOS DEL TEMA

**Objetivo 1:** *Comprender la estructura interna y la función de los multiplexos en su doble uso como:*

- a) *Selector de canales de entrada (1 de n, paso de información de paralelo a serie)*
- b) *Módulo de diseño de otros circuitos en lógica combinatorial.*

**Objetivo 2:** *Comprender los demultiplexos y su uso como:*

- a) *Selector de canales de salida (1 de n, paso de información de serie a paralelo)*
- b) *Módulo de diseño (junto con los Multiplexos) de circuitos convertidores de código.*

**Objetivo 3:** *Conocer los circuitos encargados de adaptar las señales en su interacción con un "bus": amplificadores (buffers-drivers) y transmisores-receptores de bus (bus-transceivers).*

# 6.1 Circuitos selectores de datos (Multiplexor)

- Multiplexor = circuito con “N” entradas, 1 salida y “n” patillas de selección, tal que  $2^n=N$ .
- Con la combinación binaria introducida en las patillas de selección (n), elegimos la entrada N que aparecerá en la salida.

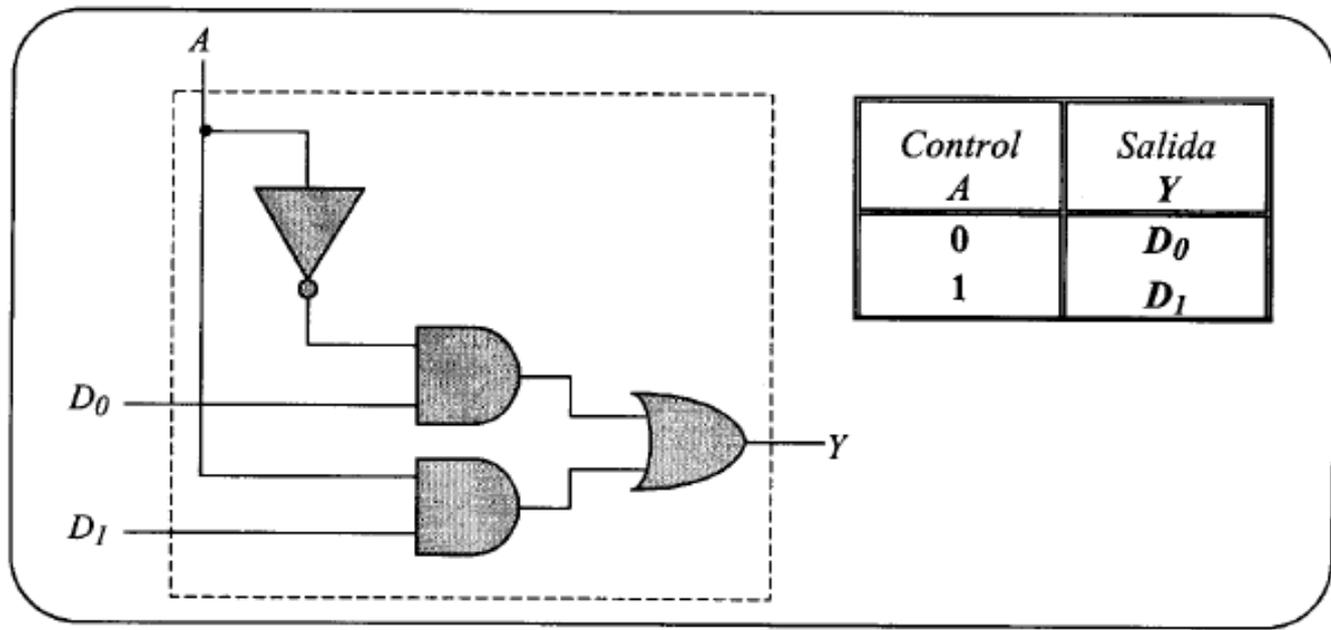
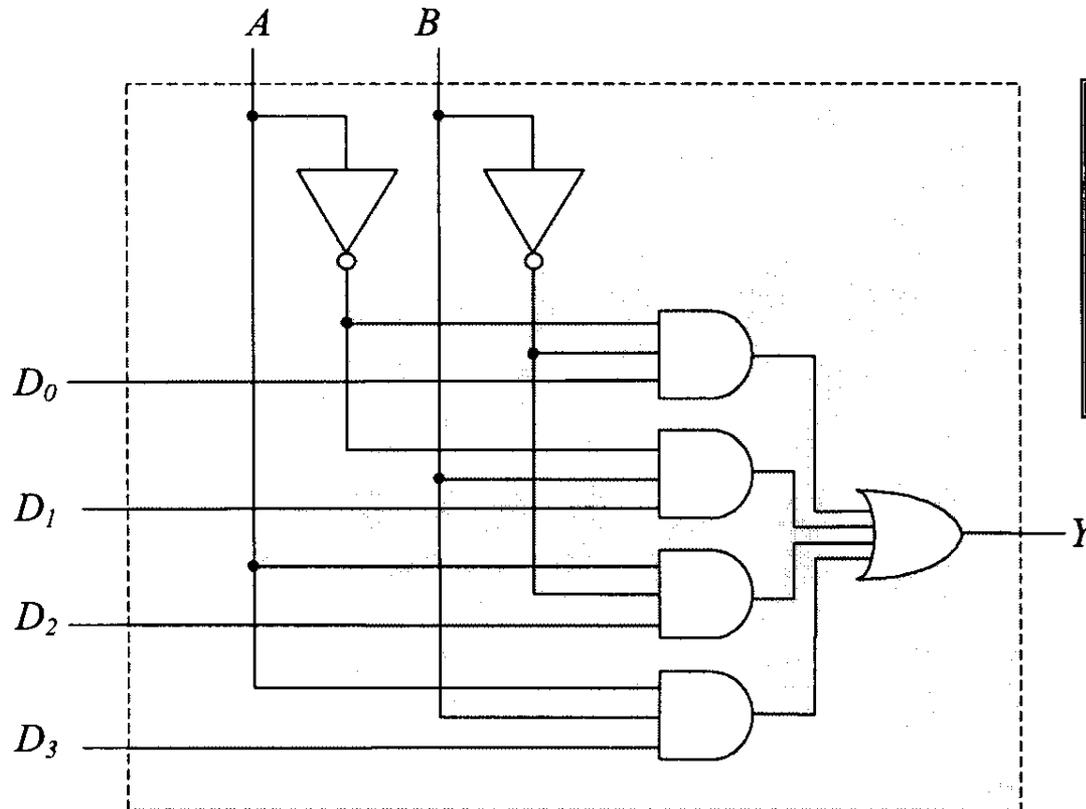


Figura 6.1. Circuito multiplexor de 2 a 1.

# Multiplexor de 4 canales

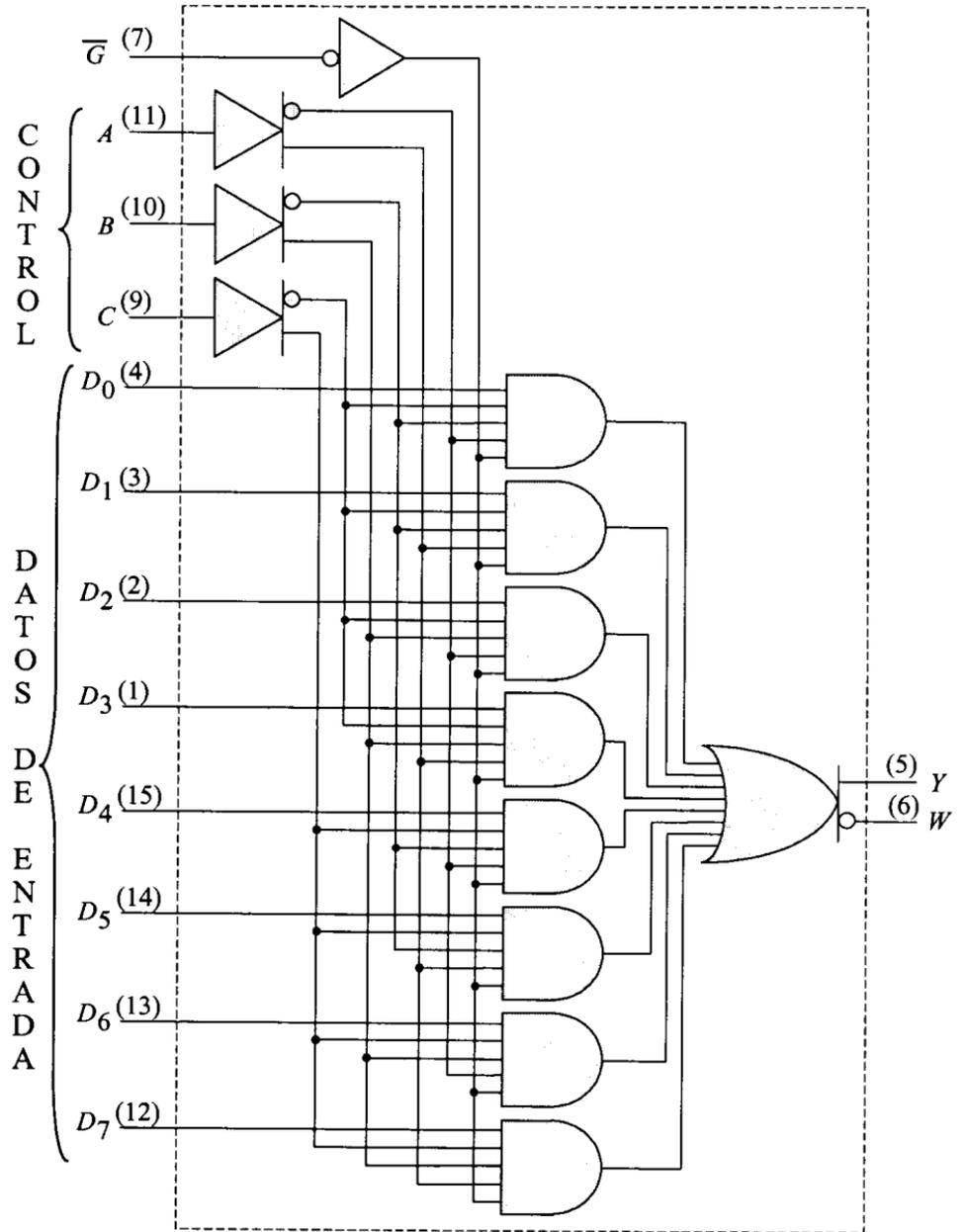
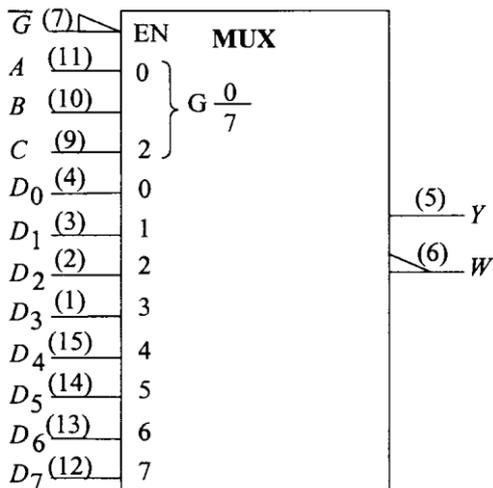


<i>Control</i>		<i>Salidas</i>
<i>A</i>	<i>B</i>	<i>Y</i>
0	0	<i>D</i> <sub>0</sub>
0	1	<i>D</i> <sub>1</sub>
1	0	<i>D</i> <sub>2</sub>
1	1	<i>D</i> <sub>3</sub>

Los multiplexores suelen llevar una señal de habilitación (Strobe) que controla su funcionamiento (si está inactiva, el multiplexor no deja pasar ninguna información).

# Multiplexor de 8 canales

ENTRADAS				SALIDAS	
Control de Canal			Strobe	Y	W
C	B	A	$\bar{G}$		
x	x	x	H	L	H
L	L	L	L	$D_0$	$\overline{D_0}$
L	L	H	L	$D_1$	$\overline{D_1}$
L	H	L	L	$D_2$	$\overline{D_2}$
L	H	H	L	$D_3$	$\overline{D_3}$
H	L	L	L	$D_4$	$\overline{D_4}$
H	L	H	L	$D_5$	$\overline{D_5}$
H	H	L	L	$D_6$	$\overline{D_6}$
H	H	H	L	$D_7$	$\overline{D_7}$



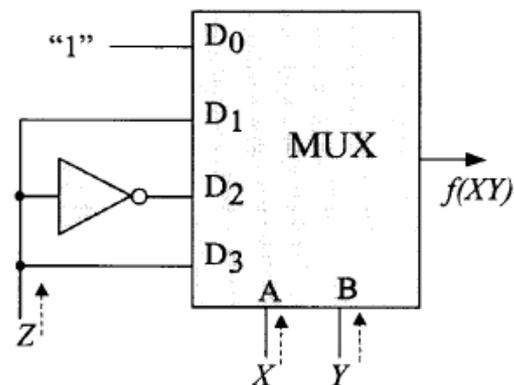
# Síntesis de funciones con multiplexores

1. Las variables de control ( $A, B$ ) representan a dos variables cualesquiera de la función a sintetizar.
2. Los canales de datos ( $D_0, D_1, D_2, D_3$ ) se usan para la tercera variable ( $Z$ ). El rango de valores para los  $D_i$  es  $(0, 1, Z, \bar{Z})$ . Si un determinado producto  $X \cdot Y$  no existe, ponemos la  $D_i$  correspondiente a "0". Si existe y no tiene  $Z$ , ponemos su  $D_i$  a "1". Si tiene  $Z$ , ponemos el valor correspondiente ( $\bar{Z}$  ó  $Z$ ).

$$f(X,Y,Z) = \bar{X}\bar{Y} + \bar{X}YZ + X\bar{Y}\bar{Z} + XYZ$$

$AB$	$XY$	Residuo	Valor de $D_i$
$m_0 = 00$	$\bar{X}\bar{Y}$	1	$D_0 = 1$
$m_1 = 01$	$\bar{X}Y$	$Z$	$D_1 = Z$
$m_2 = 10$	$X\bar{Y}$	$\bar{Z}$	$D_2 = \bar{Z}$
$m_3 = 11$	$XY$	$Z$	$D_3 = Z$

(a)



(b)

**Figura 6.5.** Diseño de funciones lógicas usando multiplexos. Ilustración del ejemplo usando un multiplexo de cuatro entradas ( $D_0, D_1, D_2$  y  $D_3$ ) y dos líneas de control ( $A$  y  $B$ ). (a) Tablas de la función a sintetizar y los valores de  $D_i$  para la variable residual. (b) Esquema de conexión del circuito.

# Síntesis con multiplexores

- El número de variables que podemos controlar con un multiplexor es el número de entradas de control + 1
- Si el número de variables es superior al del número de entradas de control + 1 del multiplexor que disponemos tendremos que hacer un diseño en árbol
  - Ejemplo: Una función de 5 variables necesita un multiplexor de 4 de control ( $4 + 1$ ) ( $2^4 = 16$  entradas).
- Si solo disponemos de multiplexores de 2 entradas (2 de control + 1 = solo tres variables), tendremos que hacer un diseño en árbol.
  - 1. Dos de las variables de la función se introducen por las entradas de las variables de control del multiplexor.
  - 2. Los canales de entrada de datos se utilizan para introducir la tercera variable.
    - Primer nivel tres variables, y las dos restantes en el segundo nivel
- Para ***ello sacamos factor común de las dos variables del vamos a emplear en el segundo nivel***

$$f(X,Y,Z,U,V) = \bar{X}\bar{Y}\bar{Z}\bar{U}V + \bar{X}\bar{Y}U\bar{V} + X\bar{Y}ZU\bar{V} + X\bar{Y}\bar{Z}U\bar{V} + XYZUV + XY\bar{Z}\bar{U}\bar{V} \quad [6.3]$$

**Paso 1º:** Identificar los términos mínimos de (X,Y) y sacar factor común.

$$f(X,Y,Z,U,V) = \bar{X}\bar{Y} \overset{D_0}{\left(\bar{Z}\bar{U}V + U\bar{V}\right)} + \bar{X}Y \overset{D_1}{(0)} + X\bar{Y} \overset{D_2}{\left(ZU\bar{V} + \bar{Z}U\bar{V}\right)} + XY \overset{D_3}{\left(ZUV + \bar{Z}\bar{U}\bar{V}\right)} \quad [6.4]$$

Así, ya tenemos el *segundo nivel*.

**Paso 2º:** Sintetizar los paréntesis en el primer nivel. Síntesis de  $(\bar{Z}\bar{U}V + U\bar{V})$ . Asociamos U, V a A, B y dejamos Z para las  $D_i$ :

$$\left(\bar{Z}\bar{U}V + U\bar{V}\right) = \bar{U}\bar{V} \overset{D_0}{(0)} + \bar{U}V \overset{D_1}{(\bar{Z})} + U\bar{V} \overset{D_2}{(1)} + UV \overset{D_3}{(0)} \quad [6.5]$$

Análogamente,

$$\left(ZU\bar{V} + \bar{Z}U\bar{V}\right) = \bar{U}\bar{V} \overset{D_0}{(0)} + \bar{U}V \overset{D_1}{(0)} + U\bar{V} \overset{D_2=1}{(Z + \bar{Z})} + UV \overset{D_3}{(0)} \quad [6.6]$$

$$\left(ZUV + \bar{Z}\bar{U}\bar{V}\right) = \bar{U}\bar{V} \overset{D_0}{(\bar{Z})} + \bar{U}V \overset{D_1}{(0)} + U\bar{V} \overset{D_2}{(0)} + UV \overset{D_3}{(Z)} \quad [6.7]$$

- Ejemplo: circuito para la función en 5 variables

- Segundo nivel

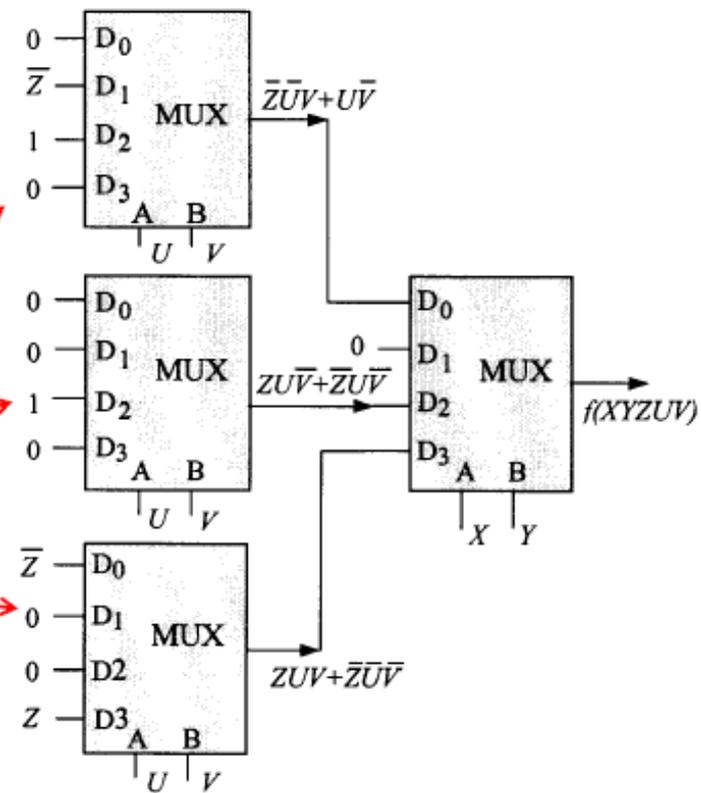
$$f(x,y,z,u,v) = \overline{x}\overline{y}(\overline{z}\overline{u}v + u\overline{v}) + \overline{x}y(0) + x\overline{y}(z\overline{u}\overline{v} + \overline{z}u\overline{v}) + xy(zuv + \overline{z}\overline{u}\overline{v})$$

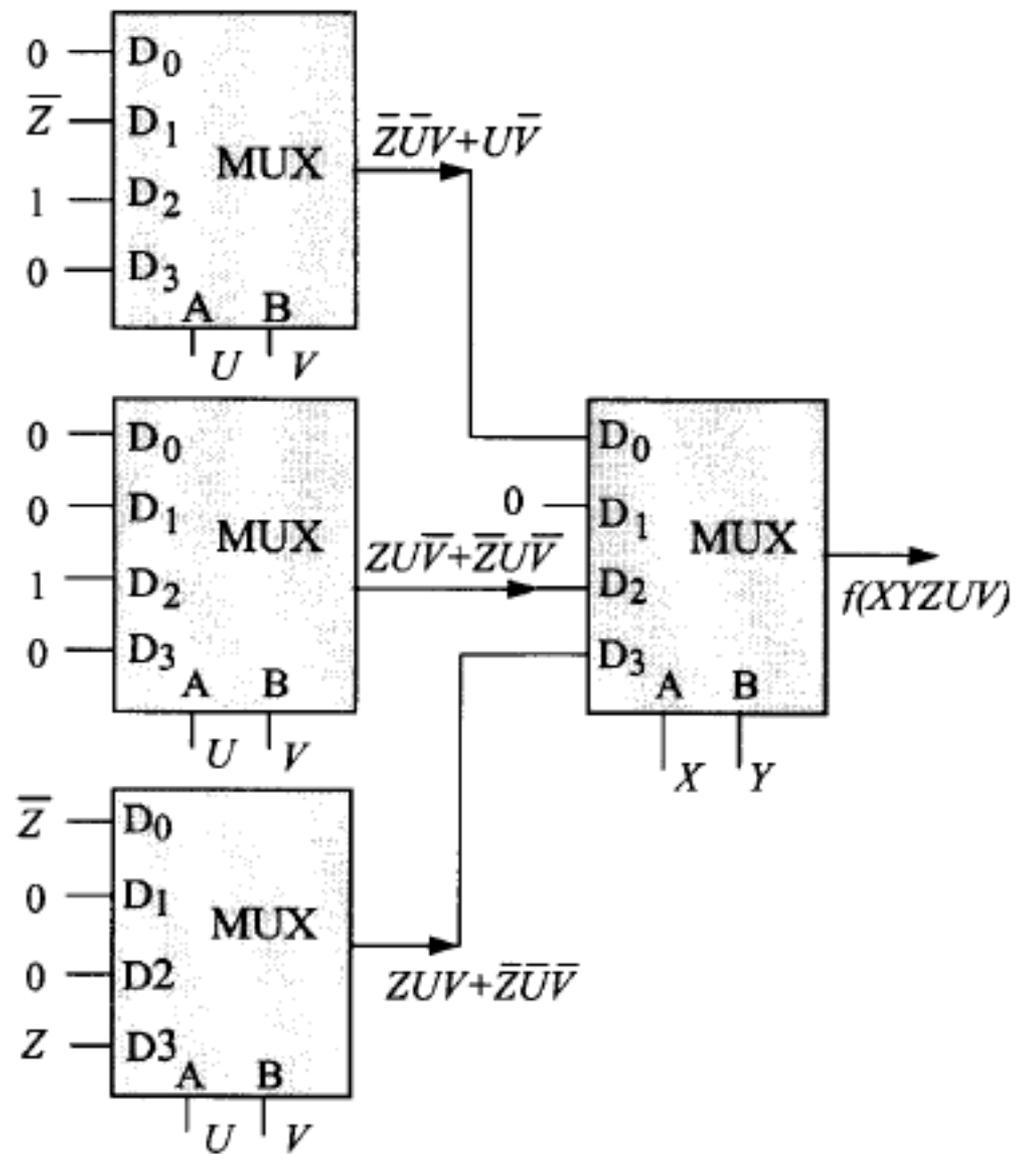
- Primer nivel

$$(\overline{z}\overline{u}v + u\overline{v}) = \overline{u}\overline{v}(0) + \overline{u}v(\overline{z}) + u\overline{v}(1) + uv(0)$$

$$(z\overline{u}\overline{v} + \overline{z}u\overline{v}) = \overline{u}\overline{v}(0) + \overline{u}v(0) + u\overline{v}(z + \overline{z}) + uv(0)$$

$$(zuv + \overline{z}\overline{u}\overline{v}) = \overline{u}\overline{v}(\overline{z}) + \overline{u}v(0) + u\overline{v}(0) + uv(z)$$

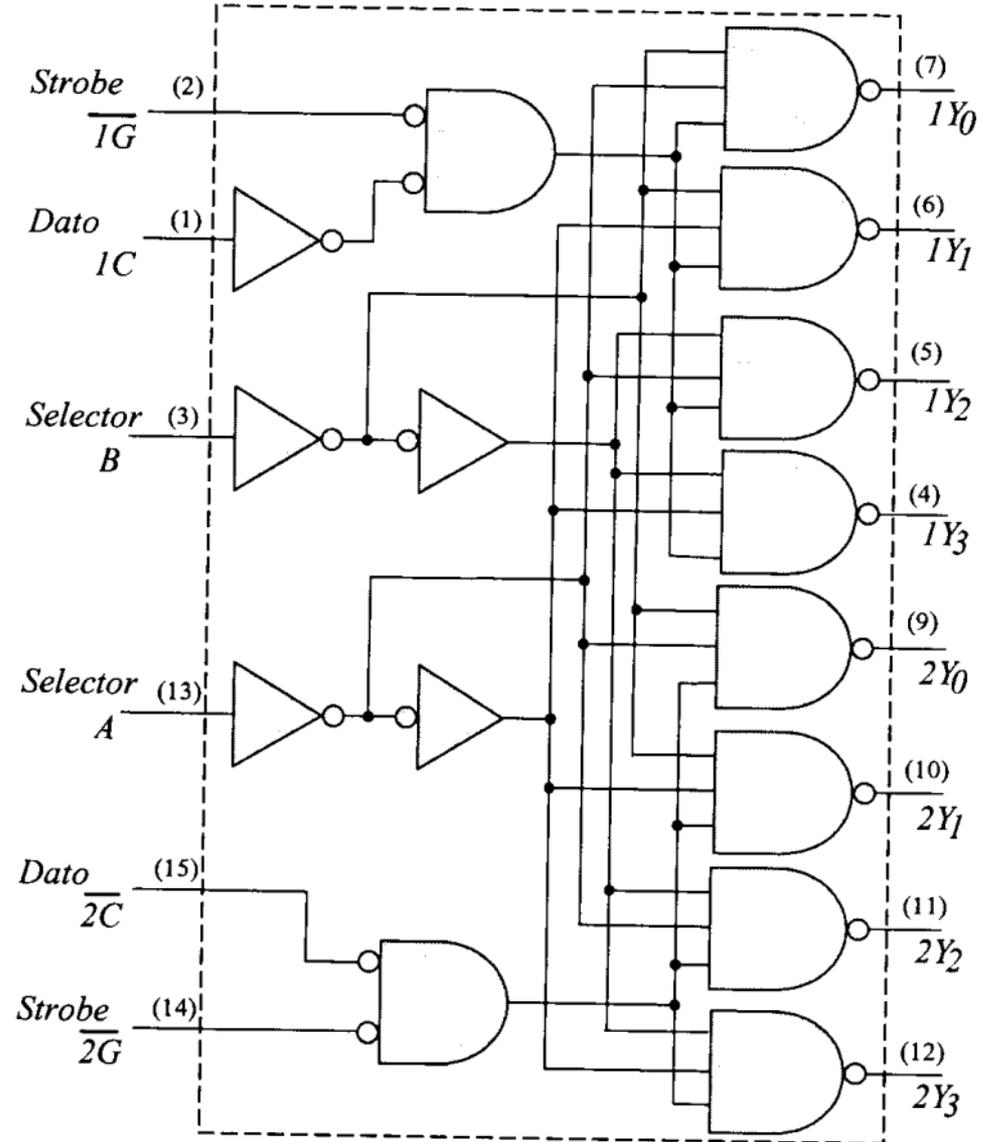




**Figura 6.6.** Diseño en dos niveles de una función de cinco variables usando MUX de cuatro entradas.

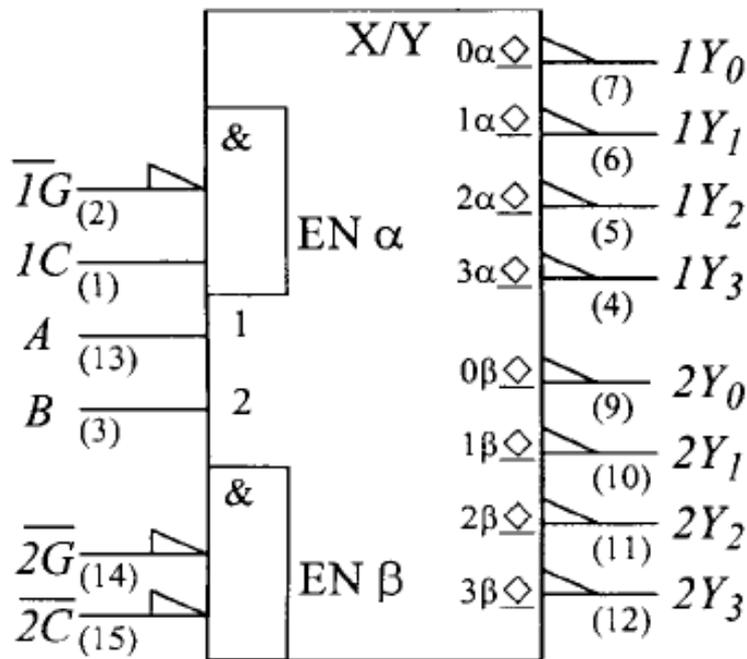
## 6.2 Demultiplexos

- Circuito con “1” entradas, N salida y “n” patillas de selección, tal que  $2^n=N$ .
- Con la combinación binaria introducida en las patillas de selección (n), elegimos la salida N en la que aparecerá la entrada.
- Ejemplo: SN74ALS156 con dos demultiplexores de 1 a 4 con direccionamiento común, entradas de habilitación individual y salidas activas por nivel bajo.

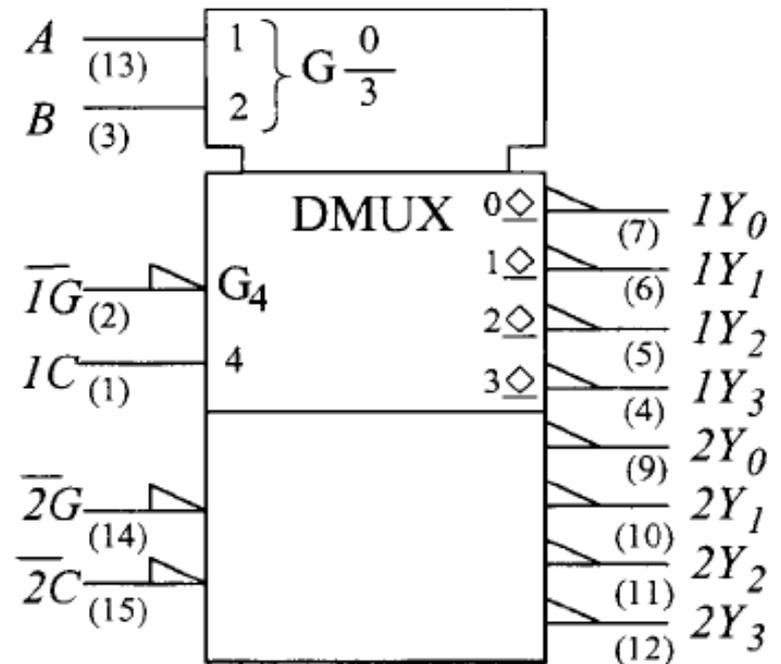


# Demultiplexores y decodificadores

- Los demultiplexores son equivalentes a los decodificadores
  - **Decodificador:** circuito con  $n$  entradas y  $2^n$  salidas, de modo que en cada momento se activa la salida que corresponde con la combinación binaria de las entradas.
- Ejemplo: SN74ALS156 como demultiplexor y como decodificador.



Decodificador de 2 a 4



Demultiplexo de 1 a 4

# Demultiplexores y decodificadores

- Ejemplo: tablas de verdad del SN74ALS156.

ENTRADAS				SALIDAS			
Selección		Strobe	Dato				
B	A	$\overline{1G}$	1C	1Y <sub>0</sub>	1Y <sub>1</sub>	1Y <sub>2</sub>	1Y <sub>3</sub>
x	x	H	x	H	H	H	H
L	L	L	H	L	H	H	H
L	H	L	H	H	L	H	H
H	L	L	H	H	H	L	H
H	H	L	H	H	H	H	L
X	X	X	L	H	H	H	H

ENTRADAS				SALIDAS			
Selección		Strobe	Dato				
B	A	$\overline{2G}$	$\overline{2C}$	2Y <sub>0</sub>	2Y <sub>1</sub>	2Y <sub>2</sub>	2Y <sub>3</sub>
x	x	H	x	H	H	H	H
L	L	L	L	L	H	H	H
L	H	L	L	H	L	H	H
H	L	L	L	H	H	L	H
H	H	L	L	H	H	H	L
X	X	X	H	H	H	H	H

- Entrada 1C activa por nivel alto.
- Entrada  $\overline{2C}$  activa por nivel bajo.
- Los decodificadores de **n** a **2<sup>n</sup>** sirven para implementar funciones de **n** variables:
  - Cada salida del decodificador representa un minterm, y basta con poner una OR que reciba como entradas los minterms de la FND.

Así pues, los circuitos demultiplexores/decodificadores intervienen en el diseño lógico a tres niveles:

- a) *Funciones específicas de distribución de datos.*
- b) *Funciones específicas de decodificación.*
- c) *Como módulos generales de diseño.*

DESCRIPCIÓN	TIPO de SALIDA	NOMBRE (SN74xxx)
DEMUX de 4 a 16	Tres estados	154
4 a 16	Colector abierto	159
4 a 10 (BCD a Decimal)	Dos estados	42
4 a 10 (Exceso 3 a Decimal)	Dos estados	43
3 a 8 con biestables en las tres líneas de dirección	Dos estados	137
Dual 2 a 4	Dos estados	139
BCD a Decimal	Colector abierto	141
BCD a Siete Segmentos	" "	246

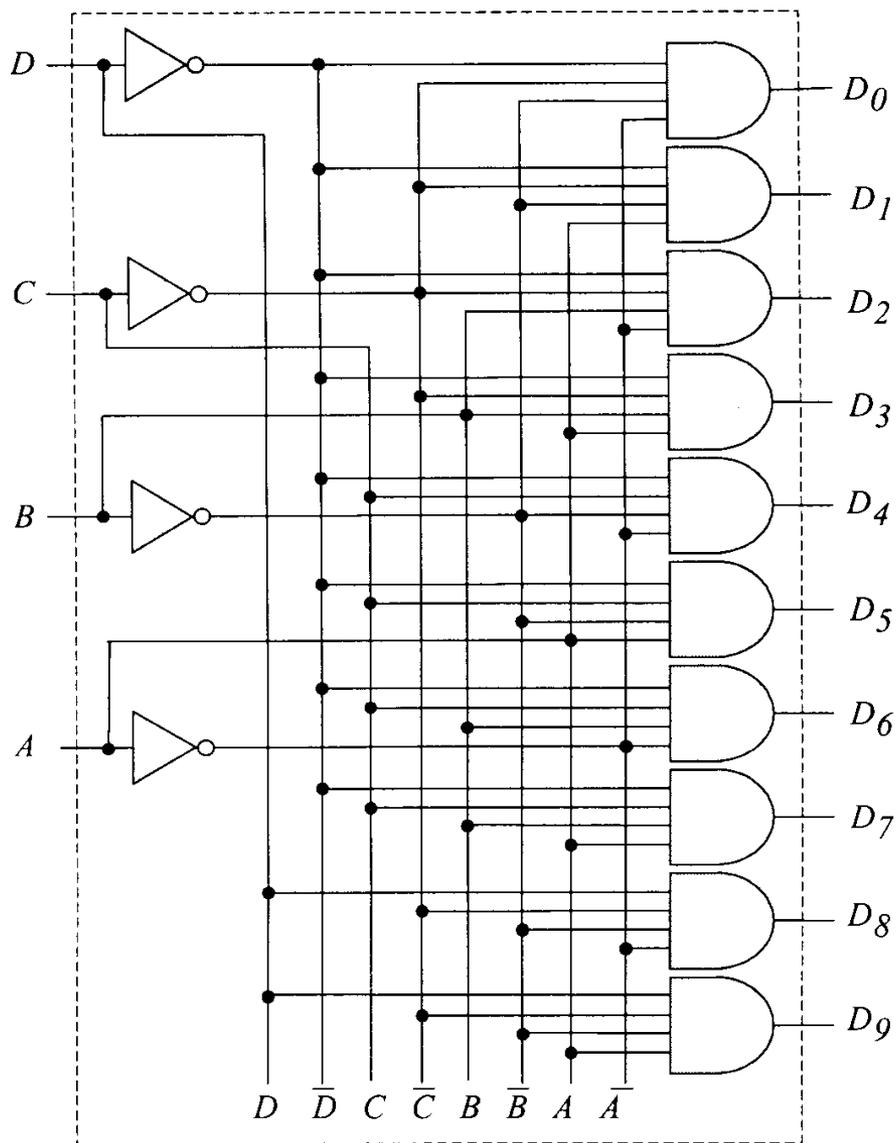
*Figura 6.10.* Algunos ejemplos de circuitos demultiplexos y decodificadores específicos.

# Uso como decodificador BCD a decimal

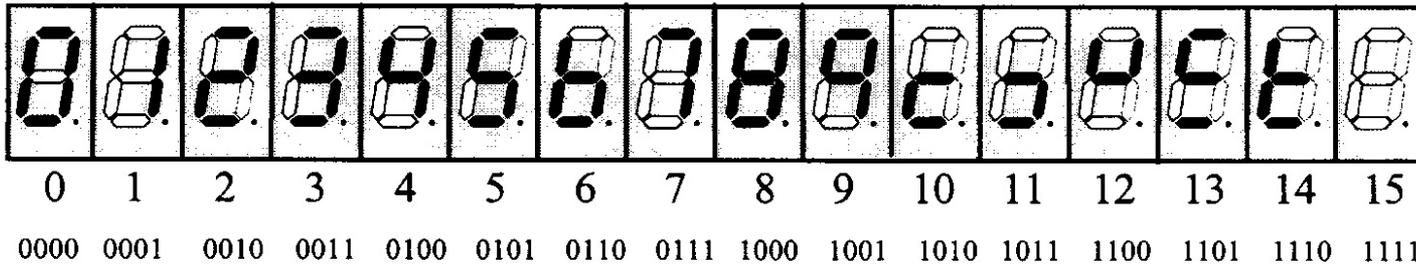
- BCD (*binary coded decimal*): sistema de numeración decimal en binario, que representa **cada dígito decimal mediante 4 dígitos binarios**.
  - Sólo son válidos los códigos del 0000 (0) al 1001 (9).
- El uso es muy sencillo, ya que, se pondrá la entrada a un nivel lógico fijo (1 o 0 según interese) y
- Luego mediante la combinación binaria que metamos en las patillas de selección conseguiremos el nivel lógico introducido a la entrada en la salida seleccionada, conformando con ello un decodificador.

Un decodificador de BDC a decimal necesita 4 líneas de entrada (A,B,C,D) y diez líneas de salida

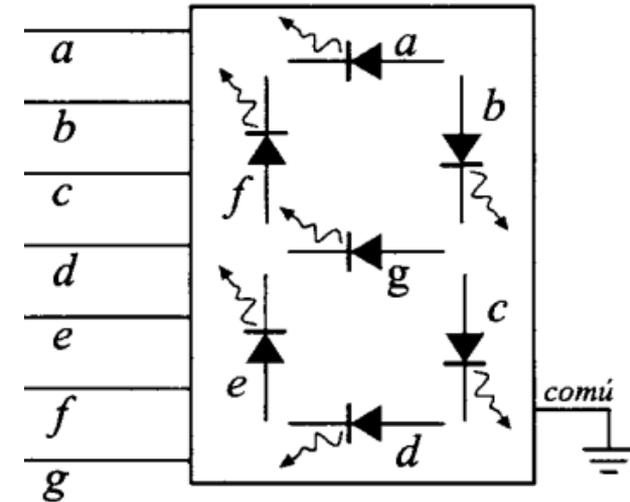
CÓDIGO BCD				DECIMAL
8	4	2	1	
<i>D</i>	<i>C</i>	<i>B</i>	<i>A</i>	$\{D_i\}$
0	0	0	0	$D_0 = \overline{D} \overline{C} \overline{B} \overline{A}$
0	0	0	1	$D_1 = \overline{D} \overline{C} \overline{B} A$
0	0	1	0	$D_2 = \overline{D} \overline{C} B \overline{A}$
0	0	1	1	$D_3 = \overline{D} \overline{C} B A$
0	1	0	0	$D_4 = \overline{D} C \overline{B} \overline{A}$
0	1	0	1	$D_5 = \overline{D} C \overline{B} A$
0	1	1	0	$D_6 = \overline{D} C B \overline{A}$
0	1	1	1	$D_7 = \overline{D} C B A$
1	0	0	0	$D_8 = D \overline{C} \overline{B} \overline{A}$
1	0	0	1	$D_9 = D \overline{C} \overline{B} A$



# Decodificador BCD a 7 segmentos

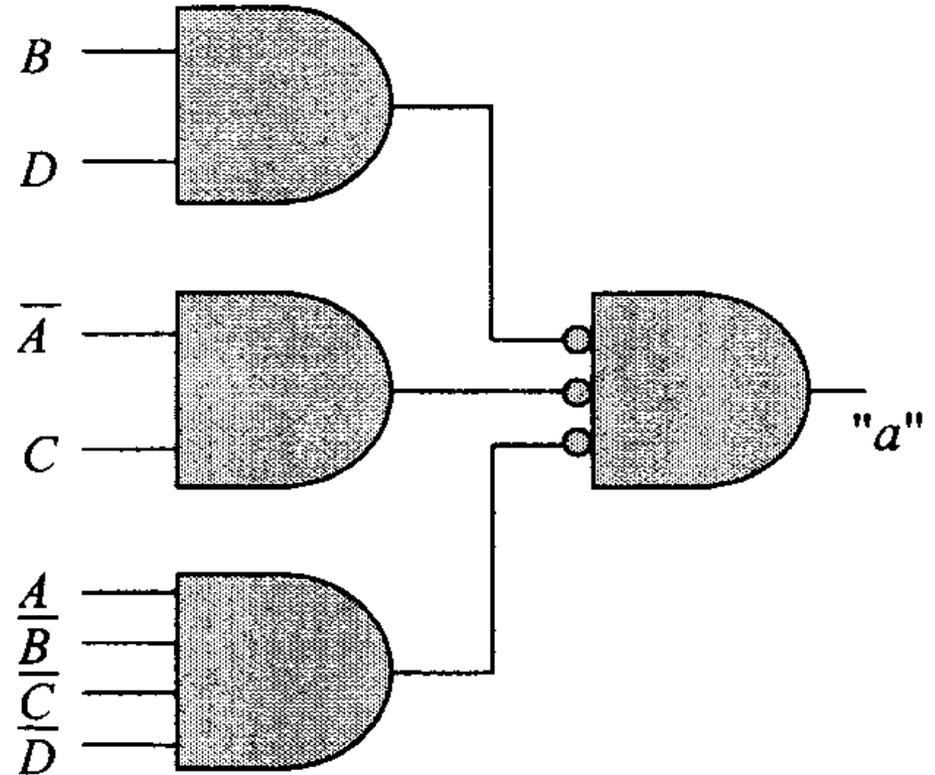


	<i>D</i>	<i>C</i>	<i>B</i>	<i>A</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	0	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	0	1	1
10	1	0	1	0	0	0	0	1	1	0	1
11	1	0	1	1	0	0	1	1	0	0	1
12	1	1	0	0	0	1	0	0	0	1	1
13	1	1	0	1	1	0	0	1	0	1	1
14	1	1	1	0	0	0	0	1	1	1	1
15	1	1	1	1	0	0	0	0	0	0	0



# Síntesis del segmento "a"

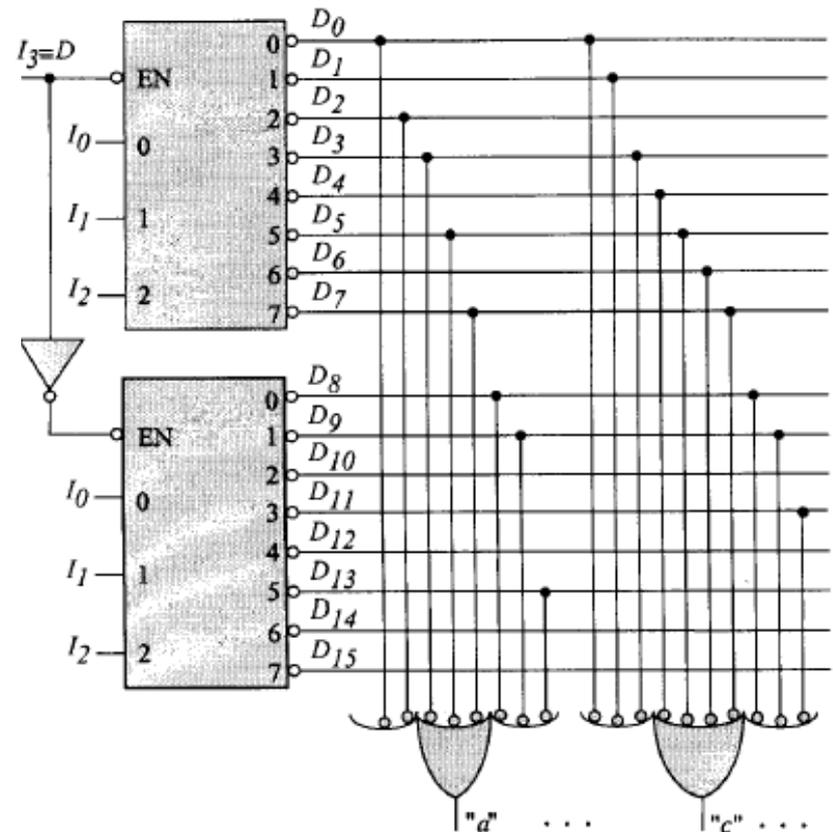
<i>DC</i> \ <i>BA</i>	<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>00</i>	1	0	1	1
<i>01</i>	0	1	1	0
<i>11</i>	0	1	0	0
<i>10</i>	1	1	0	0



# Decodificadores como módulos universales de síntesis de funciones lógicas

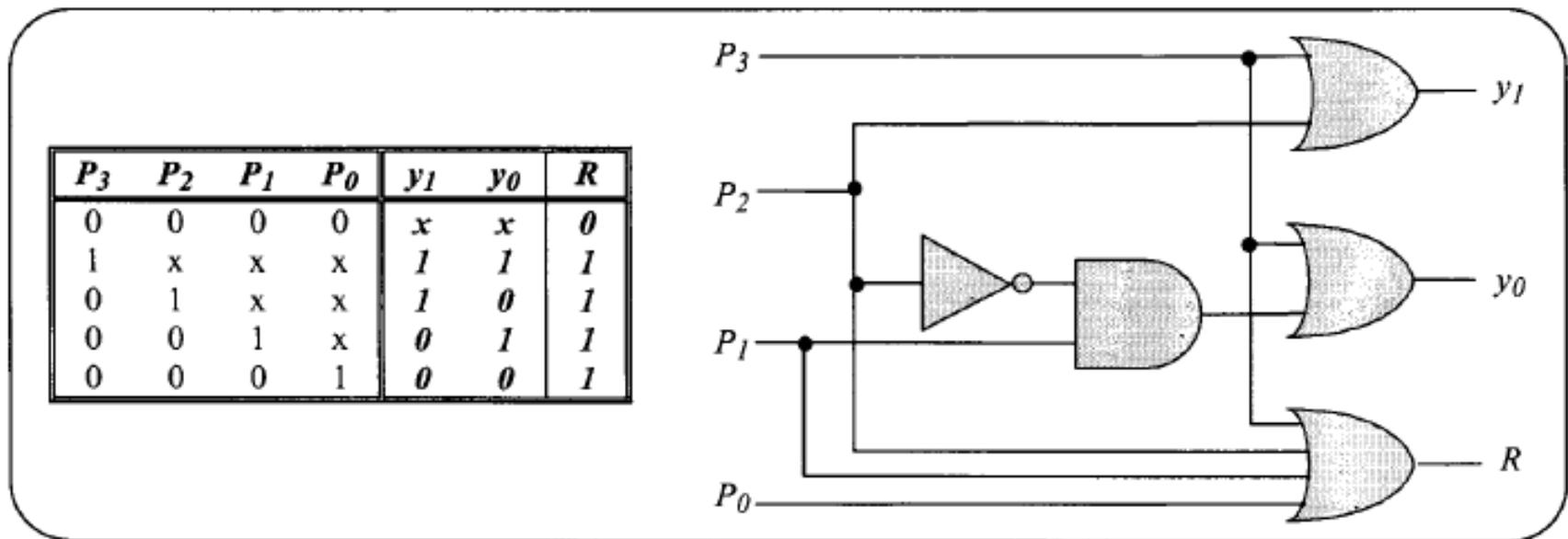
- Decodificador de BCD a 7 segmentos mediante dos SN74138 (decodificador de 3 a 8):
- La entrada de datos más significativa la usamos para introducirla por las respectivas entradas de *Strobe* (una afirmada y la otra negada):

	D	C	B	A	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	0	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	0	1	1
10	1	0	1	0	0	0	0	1	1	0	1
11	1	0	1	1	0	0	1	1	0	0	1
12	1	1	0	0	0	1	0	0	0	1	1
13	1	1	0	1	1	0	0	1	0	1	1
14	1	1	1	0	0	0	0	1	1	1	1
15	1	1	1	1	0	0	0	0	0	0	0



## 6.3 Codificadores con prioridad

- Un circuito codificador genera una salida en función del código correspondiente a la entrada activa.
- Si están activadas más de una entrada es necesario establecer un criterio de prioridad de forma que en todo momento sólo se genere el código de la línea más prioritaria de entre todas las activas.
- A este tipo de codificadores se les denomina codificadores con prioridad



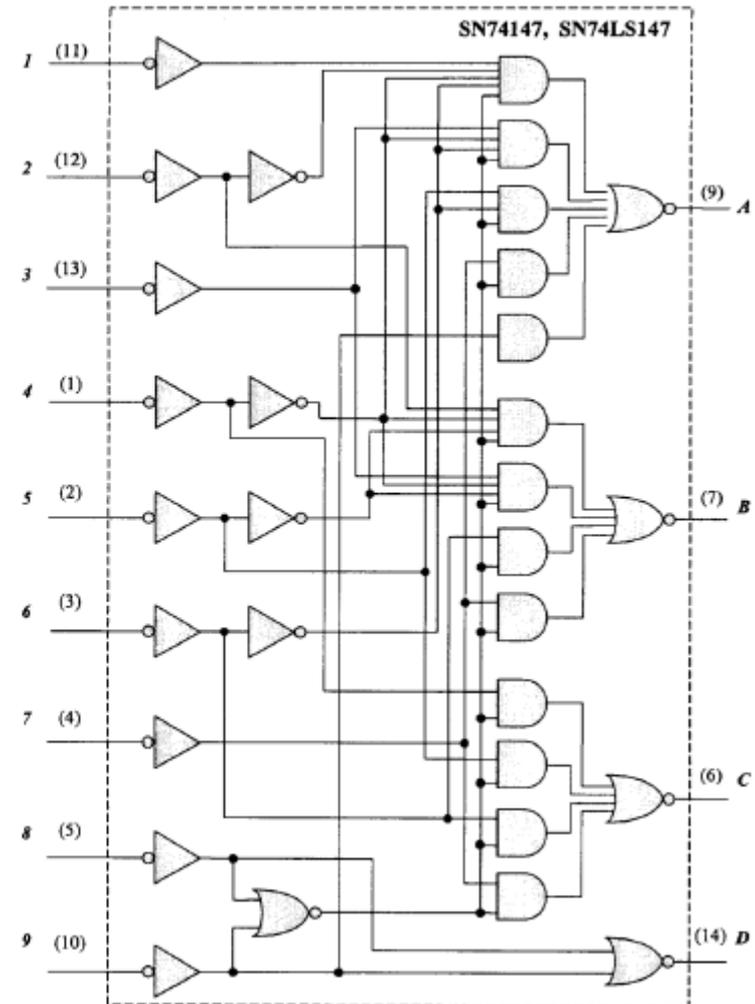
**Figura 6.16.** Tabla de verdad y circuito correspondiente al diseño de un codificador con cuatro niveles de prioridad. Se considera que  $P_3, P_2, P_1, P_0$  y  $R$  son activas en baja y que  $y_1$  e  $y_0$  son activas en alta.

# Codificadores con prioridad

- Circuito SN74147 de 9 líneas de datos en decimal a 4 líneas en BCD:

<i>Entradas</i>									<i>Salidas</i>			
1	2	3	4	5	6	7	8	9	<i>D</i>	<i>C</i>	<i>B</i>	<i>A</i>
H	H	H	H	H	H	H	H	H	<i>H</i>	<i>H</i>	<i>H</i>	<i>H</i>
x	x	x	x	x	x	x	x	L	<i>L</i>	<i>H</i>	<i>H</i>	<i>L</i>
x	x	x	x	x	x	x	L	H	<i>L</i>	<i>H</i>	<i>H</i>	<i>H</i>
x	x	x	x	x	x	L	H	H	<i>H</i>	<i>L</i>	<i>L</i>	<i>L</i>
x	x	x	x	x	L	H	H	H	<i>H</i>	<i>L</i>	<i>L</i>	<i>H</i>
x	x	x	x	L	H	H	H	H	<i>H</i>	<i>L</i>	<i>H</i>	<i>L</i>
x	x	L	H	H	H	H	H	H	<i>H</i>	<i>H</i>	<i>L</i>	<i>L</i>
x	L	H	H	H	H	H	H	H	<i>H</i>	<i>H</i>	<i>L</i>	<i>H</i>
L	H	H	H	H	H	H	H	H	<i>H</i>	<i>H</i>	<i>H</i>	<i>L</i>

- Entradas y salidas activas por nivel bajo.

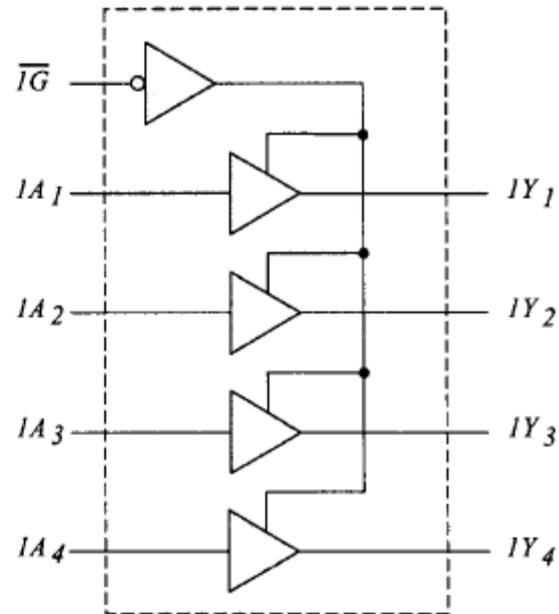


## 6.4 Amplificadores y transmisores receptores de bus

- Bus: canal de comunicación al que se conectan varios dispositivos con capacidad de poner información o leerla.
  - En un bus, no es posible que varios elementos conectados pongan información en el mismo simultáneamente, ya que se mezclarían todos los valores y no se podría leer la información correcta.
- Los amplificadores (buffers-drivers) y los transmisores-receptores de bus (bus transceivers) se emplean para permitir la conexión de varios elementos a un bus, de modo que no se produzcan interferencias eléctricas en el mismo.
- Los amplificadores y los transmisores-receptores de bus no realizan cambios lógicos en las señales que reciben, pero:
  - Las preparan para que estén disponibles donde y cuando se necesiten.
  - Dotan a las señales del nivel de corriente suficiente como para que no existan problemas eléctricos.
  - Aíslan la fuente de la señal al circuito al que debe ser aplicada dicha señal.

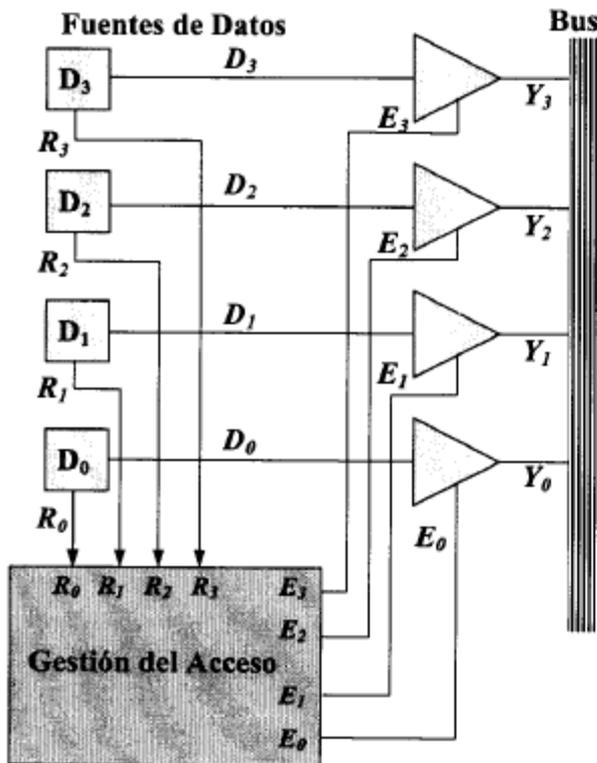
# Amplificadores (buffers-drivers)

- Un **buffer-driver** es un amplificador digital que incrementa o absorbe el nivel de corriente de una señal sin confundir el estado.
  - Son de tipo unidireccional.
  - Proporcionan salida triestado: alto, bajo o desconexión (alta impedancia).
- Si la señal de habilitación está inactiva: desconexión.
- Si la señal de control está activa: sale el valor que hay en la entrada de datos, regenerado eléctricamente, pero sin cambiar de estado.
  - Se usan para conectar dispositivos a buses.
- Ejemplo de buffer-driver: 74ALS760 (medio circuito)
  - Cada medio circuito permite conectar a un bus una fuente de un dato de 4 bits.
  - Si la señal de *strobe*  $1G$  está desactivada (nivel alto), las salidas se encuentran en estado de desconexión (alta impedancia).
  - Si la señal  $1G$  está activada (nivel bajo), cada salida  $Y_i$  presenta el valor lógico de su entrada  $A_i$ .



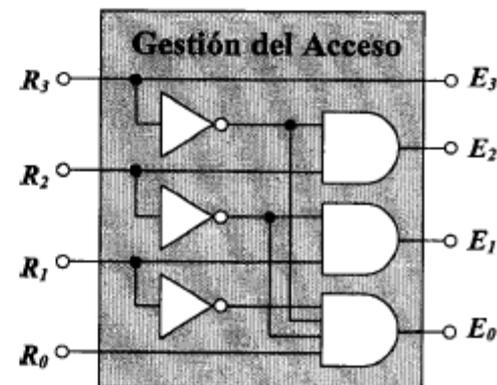
# Amplificadores (buffers-drivers)

- Ejemplo: queremos diseñar un circuito que controle el acceso a un bus de 4 fuentes de señal D0, D1, D2 y D3, de un bit cada una, de modo que:
  - Cada fuente incluye un bit adicional Ri que se pone en alta cada vez que se desea acceder al bus.
  - Si varias fuentes quieren acceder al bus a la vez, la prioridad es D3 > D2 > D1 > D0.



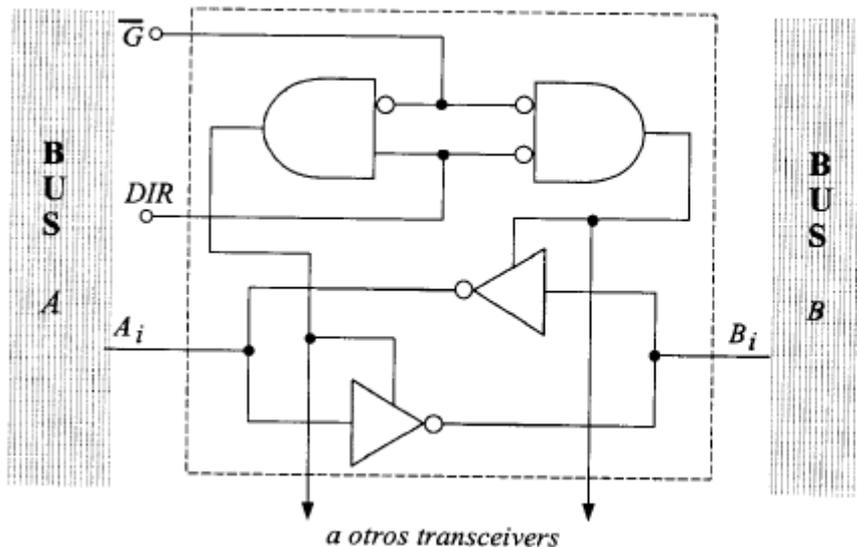
R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	E <sub>3</sub>	E <sub>2</sub>	E <sub>1</sub>	E <sub>0</sub>
1	x	x	x	1	0	0	0
0	1	x	x	0	1	0	0
0	0	1	x	0	0	1	0
0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0

$E_3 = R_3$ ,  $E_2 = \bar{R}_3 R_2$ ,  $E_1 = \bar{R}_3 \bar{R}_2 R_1$ ,  $E_0 = \bar{R}_3 \bar{R}_2 \bar{R}_1 R_0$



# Transmisores-receptores de canal

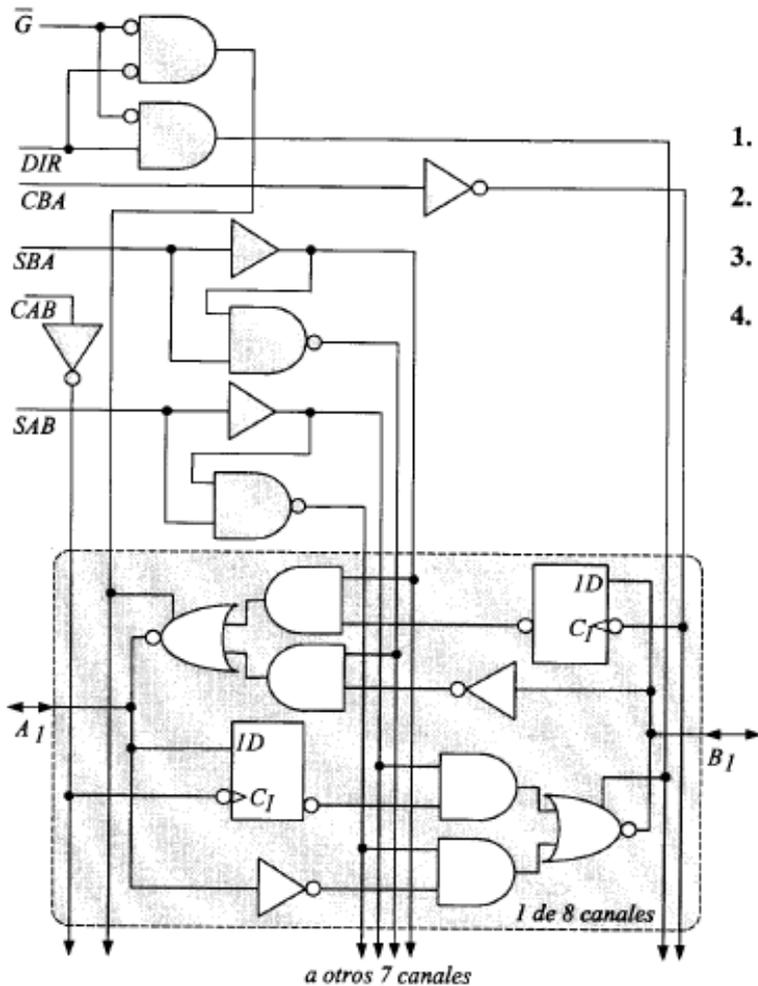
- Los transmisores-receptores de canal (bus transceivers):
  - Son de tipo bidireccional.
  - Mantienen las características de los drivers para cada dirección de transmisión.
  - Duplican el circuito básico e incluyen dos señales de control ( $DIR$  y  $G$ ) que facilitan la transmisión en un sentido al mismo tiempo que la inhiben en sentido contrario.
- Los transceivers están pensados para facilitar las comunicaciones asíncronas y bidireccionales entre dos canales de datos (buses) A y B.
  - Ejemplo: 74ALS641



CONTROL		OPERACIÓN
$\overline{G}$	$DIR$	
0	0	Datos de B pasan a A
0	1	Datos de A pasan a B
1	x	Aislamiento

# Transmisores-receptores de canal

- Ejemplo: octal bus transceiver 74ALS646.



## Funciones:

- Paso directo del canal *B* al canal *A* en tiempo real.
- " " " " *A* " " *B* " " "
- Almacenamiento en los registros internos de datos procedentes del *A*, del *B* o de ambos.
- Transferencia de datos almacenados previamente a los canales *A* ó *B*.

ENTRADAS						DATOS I/O		FUNCIÓN
$\bar{G}$	<i>DIR</i>	<i>CAB</i>	<i>CBA</i>	<i>SAB</i>	<i>SBA</i>	<i>A1-A8</i>	<i>B1-B8</i>	<i>ALS646,647</i>
x	x	↑	x	x	x	Entrada	NE	Almacena <i>A</i> , <i>B</i> , NE
x	x	x	↑	x	x	NE	Entrada	Almacena <i>B</i> , <i>A</i> , NE
H	x	↑	↑	x	x	Entrada	Entrada	Almacena <i>A</i> y <i>B</i>
H	x	H/L	H/L	x	x	"	"	Aislamiento (HS)
L	L	x	x	x	L	Salida	Entrada	Datos bus <i>B</i> al <i>A</i> (TR)
L	L	x	H/L	x	H	"	"	Datos <i>B</i> a bus <i>A</i> (ST)
L	H	x	x	L	x	Entrada	Salida	Datos <i>A</i> a bus <i>B</i> (TR)
L	H	H/L	x	H	x	"	"	Datos <i>A</i> a bus <i>B</i> (ST)

CBA y CAB: señales de reloj.

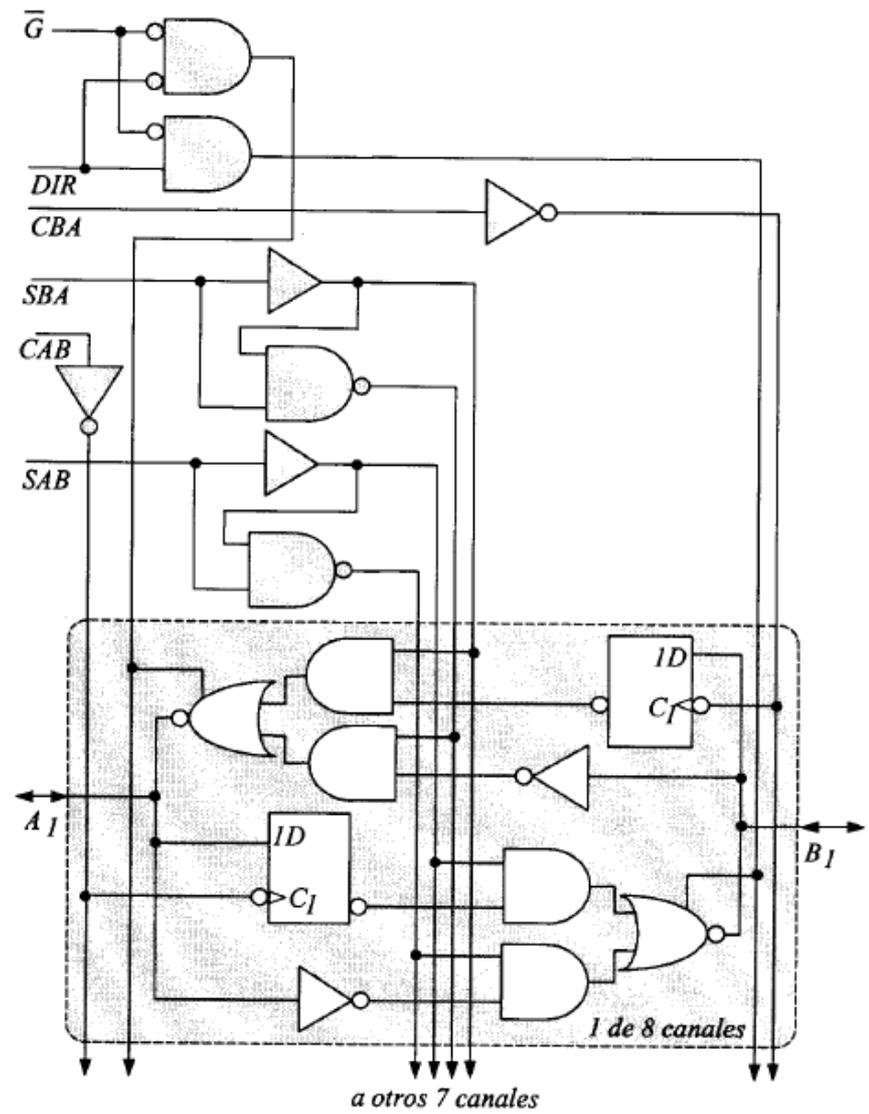
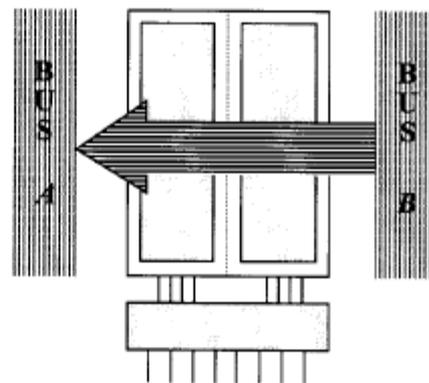


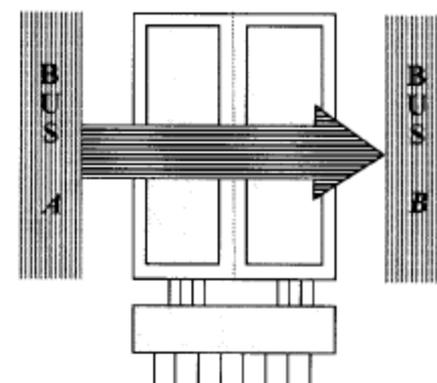
Figura 6.22. Circuito interno del octal bus transceiver ALS646

- Ejemplo: modos de funcionamiento del 74ALS646.

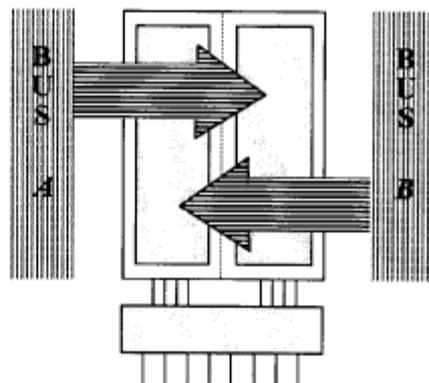
ENTRADAS						DATOS I/O		FUNCIÓN
$\overline{G}$	DIR	CAB	CBA	SAB	SBA	A1-A8	B1-B8	ALS646,647
x	x	↑	x	x	x	Entrada	NE	Almacena A, B, NE
x	x	x	↑	x	x	NE	Entrada	Almacena B, A, NE
H	x	↑	↑	x	x	Entrada	Entrada	Almacena A y B
H	x	H/L	H/L	x	x	"	"	Aislamiento (HS)
L	L	x	x	x	L	Salida	Entrada	Datos bus B al A (TR)
L	L	x	H/L	x	H	"	"	Datos B a bus A (ST)
L	H	x	x	L	x	Entrada	Salida	Datos A a bus B (TR)
L	H	H/L	x	H	x	"	"	Datos A a bus B (ST)



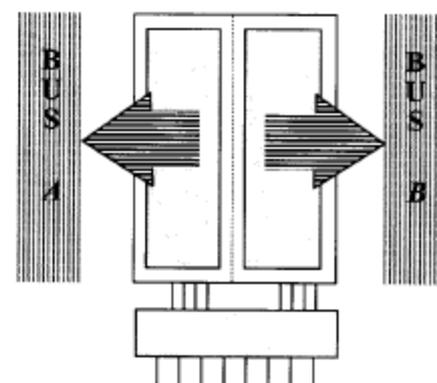
$\overline{G}$	DIR	CAB	CBA	SAB	SBA
L	L	x	x	x	L



$\overline{G}$	DIR	CAB	CBA	SAB	SBA
L	H	x	x	L	x



$\overline{G}$	DIR	CAB	CBA	SAB	SBA
x	x	↑	x	x	x
x	x	x	↑	x	x
H	x	↑	↑	x	x

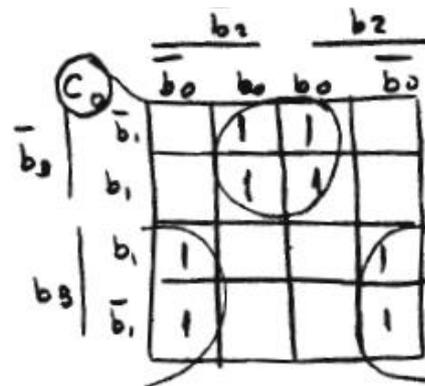


$\overline{G}$	DIR	CAB	CBA	SAB	SBA
L	L	x	H/L	x	H
L	H	H/L	x	H	x

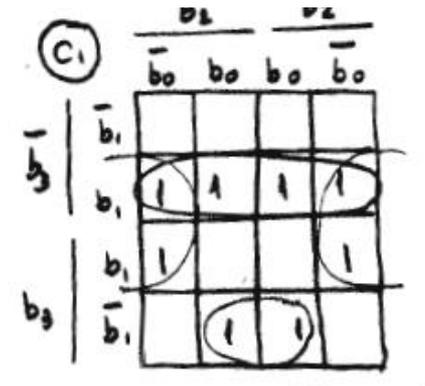
1. Diseñe un convertidor de código de 4 bits que nos permita pasar de forma automática de la representación de números positivos y negativos en Complemento a 1 (C-1) a su representación en Complemento a 2 (C-2).

1) Convertidor de Cal. a Ca2

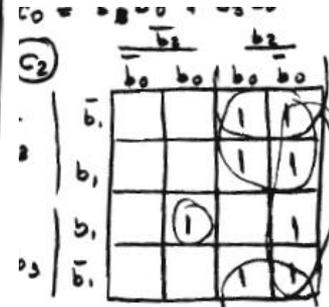
muestra	Cal				Ca2			
	$b_3$	$b_2$	$b_1$	$b_0$	$c_3$	$c_2$	$c_1$	$c_0$
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	1
4	0	1	0	0	0	1	0	0
5	0	1	0	1	0	1	0	1
6	0	1	1	0	0	1	1	0
7	0	1	1	1	0	1	1	1
8	1	0	0	0	1	0	0	1
9	1	0	0	1	1	0	1	0
10	1	0	1	0	1	0	1	1
11	1	0	1	1	1	1	0	0
12	1	1	0	0	1	1	0	1
13	1	1	0	1	1	1	1	0
14	1	1	1	0	1	1	1	1
15	1	1	1	1	0	0	0	0



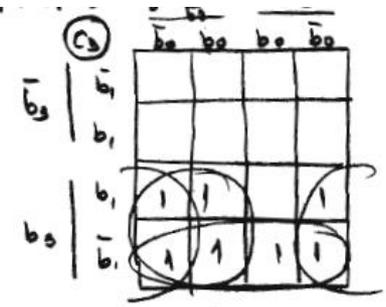
$$C_0 = \bar{b}_2 b_0 + b_2 b_1$$



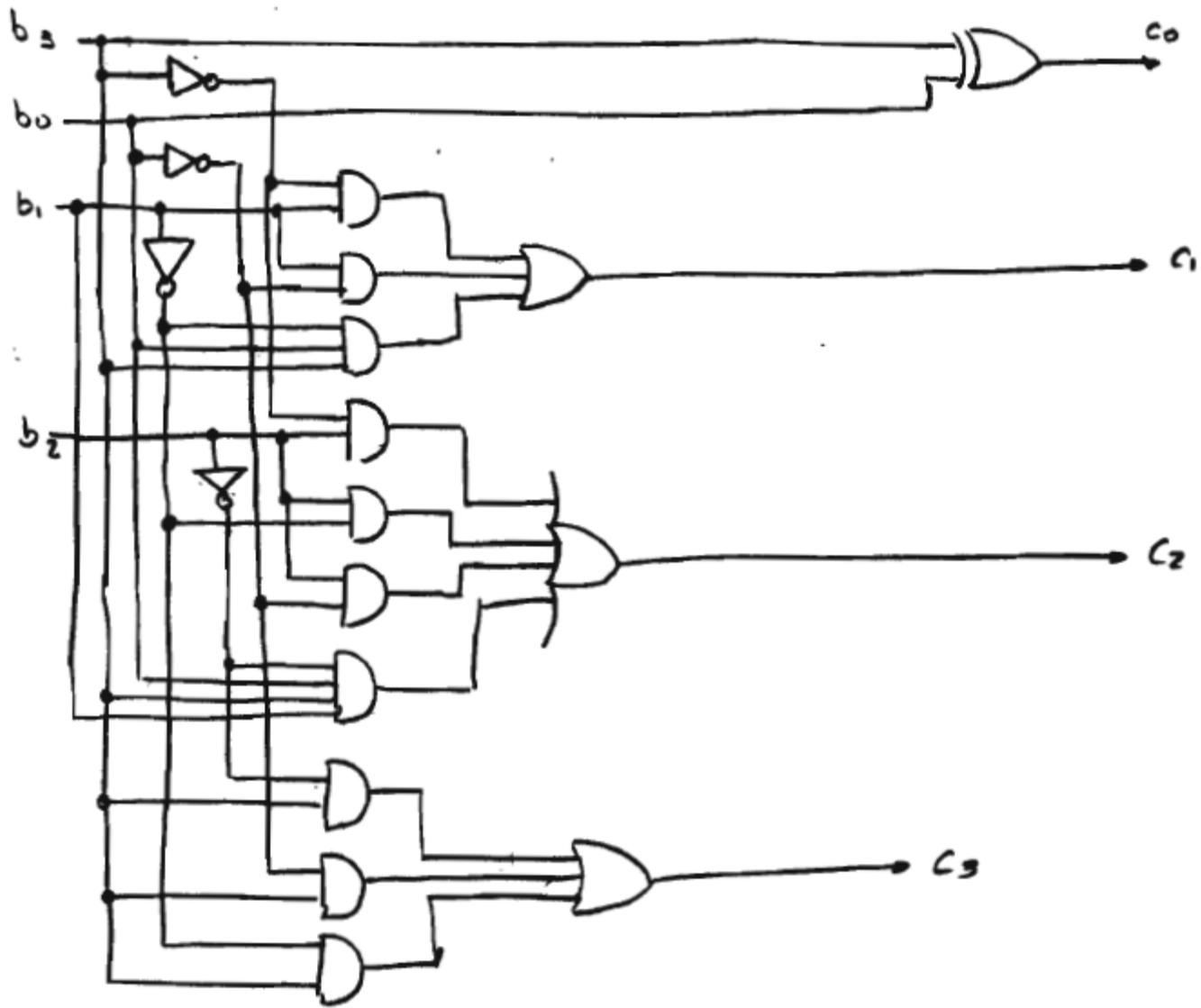
$$C_1 = \bar{b}_2 b_1 + b_1 \bar{b}_0 + b_2 b_1 b_0$$



$$C_2 = \bar{b}_3 b_2 + b_2 \bar{b}_1 + b_1 \bar{b}_0 + b_3 b_2 b_1 b_0$$



$$C_3 = b_2 b_3 + b_3 b_0 + b_3 b_1$$



$C_{a2} \Rightarrow C_{a1} + 1$  en los negativos  
en  $C_{a1}$  empiezan por "9"

$C_{a2} = C_{a1}$  en positivos  
en  $C_{a1}$  empiezan por "0"

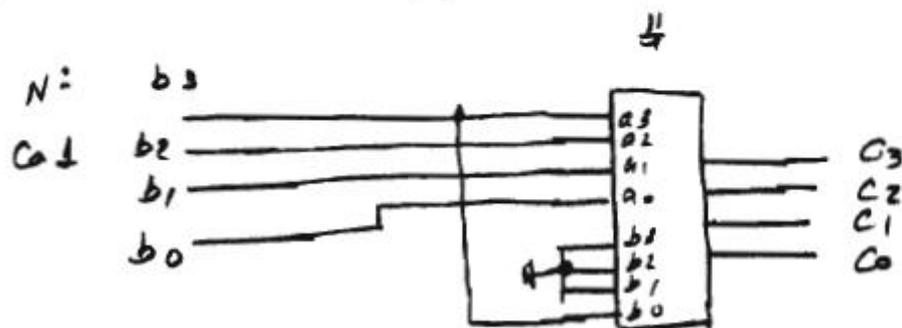
$$C_3 = \sum m(8, 9, 10, 11, 12, 13, 14)$$

$$C_2 = \sum m(4, 5, 6, 7, 11, 12, 13, 14)$$

$$C_1 = \sum m(2, 3, 6, 7, 9, 10, 13, 14)$$

$$C_0 = \sum m(1, 3, 5, 7, 8, 10, 12, 14)$$

De todas formas el mejor circuito sería un sumador al que le sumásemos el signo al  $n^2$  en  $C_{a1}$



1.- Dada la función de 2 variables,  $f = \bar{x}_0 \bar{x}_1 + x_0 x_1$ , expresada en su forma normal disyuntiva (suma de minterms) ¿cuál es la representación de esta misma función en su forma normal conjuntiva (producto de maxterms)?.

a)  $f = (x_0 + x_1) \cdot (\bar{x}_0 + \bar{x}_1)$

b)  $\bar{f} = (x_0 + \bar{x}_1) \cdot (\bar{x}_0 + x_1)$

c)  $f = (x_0 + \bar{x}_1) \cdot (\bar{x}_0 + x_1)$

d)  $f = (x_0 + x_1) \cdot (x_0 + \bar{x}_1)$

①  $f = \bar{x}_0 \bar{x}_1 + x_0 x_1$

$f = (x_0 + \bar{x}_1) (\bar{x}_0 + x_1)$   
 $\Downarrow$   
 $\subseteq$

$x_0$	$x_1$	$S$
0	0	1
0	1	0
1	0	0
1	1	1

3.- Las funciones lógicas de la diferencia y del arrastre del restador completo son:

a)  $D_i = A_i \oplus B_i \oplus C_i, \quad C_{i+1} = \bar{A}_i B_i + C_i (\overline{A_i \oplus B_i})$

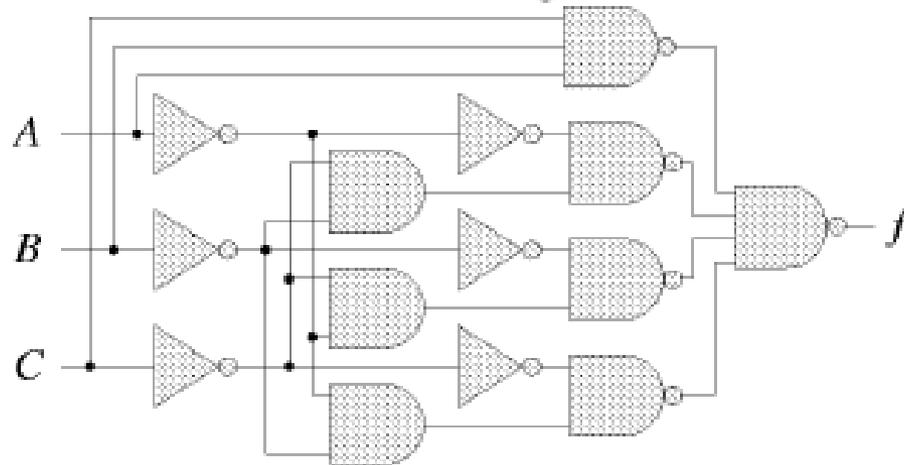
b)  $D_i = A_i \oplus B_i \oplus C_i, \quad C_{i+1} = A_i B_i + C_i (A_i \oplus B_i)$

c)  $D_i = A_i \oplus B_i \oplus \bar{C}_i, \quad C_{i+1} = A_i \bar{B}_i + C_i (\overline{A_i \oplus B_i})$

d)  $D_i = A_i \oplus B_i \oplus C_i, \quad C_{i+1} = A_i \bar{B}_i + C_i (A_i \oplus B_i)$

3<sup>a</sup> 
$$\left. \begin{aligned} D_i &= A_i \oplus B_i \oplus C_i \\ C_{i+1} &= \bar{A}_i B_i + C_i (\overline{A_i \oplus B_i}) \end{aligned} \right\} \Rightarrow \underline{a} \quad \text{Pg } 275$$

4.- ¿Cuál de las 4 soluciones dadas es la función que realiza el circuito de la figura?.



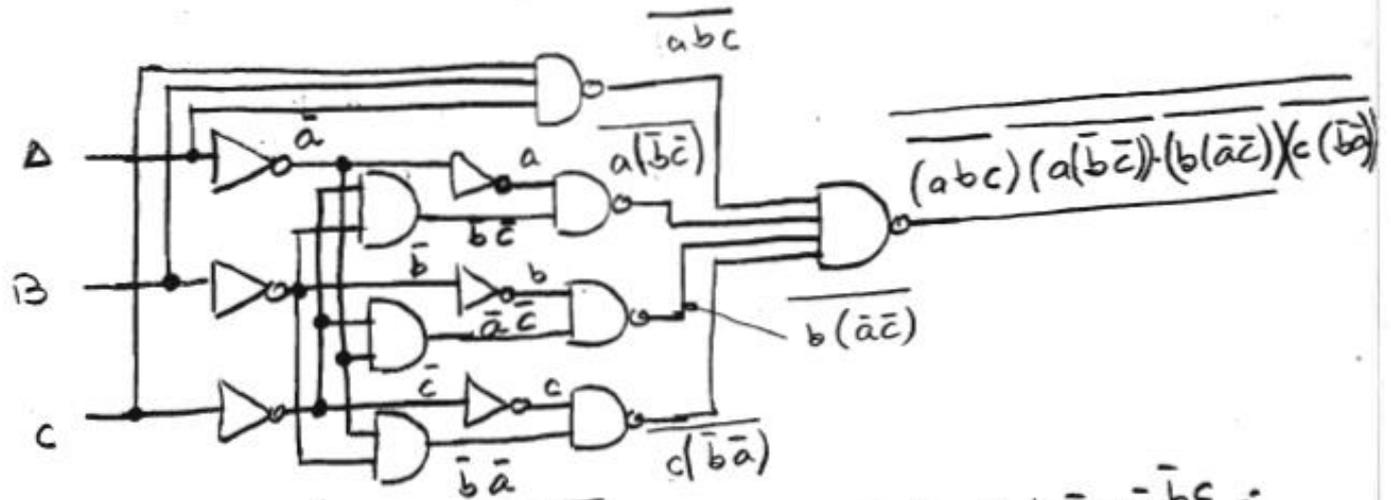
a)  $f = A \oplus B \oplus C$

b)  $f = \overline{(A B C) (A \bar{B} \bar{C}) (\bar{A} B \bar{C}) (\bar{A} \bar{B} C)}$

c)  $f = (A B C) (A \bar{B} \bar{C}) (\bar{A} B \bar{C}) (\bar{A} \bar{B} C)$

d)  $f = \overline{A \oplus B \oplus C}$

4<sup>a</sup>



$$f = \overline{abc} \overline{a\bar{b}\bar{c}} \overline{\bar{a}b\bar{c}} \overline{\bar{a}\bar{b}c} = abc + a\bar{b}\bar{c} + \bar{a}b\bar{c} + \bar{a}\bar{b}c$$

$$f = a \underbrace{(bc + \bar{b}\bar{c})}_{b \oplus c} + \bar{a} \underbrace{(b\bar{c} + \bar{b}c)}_{b \oplus c} = a \underbrace{(b \oplus c)}_m + \bar{a} \underbrace{(b \oplus c)}_m =$$

$$= a \bar{m} + \bar{a} m = a \oplus m =$$

$$f = a \oplus b \oplus c \Rightarrow \text{IP}$$

## **1. Diseño del Detector de Paridad de las palabras de entrada de 4 bits, A (A3 A2 A1 A0):**

- 1.1. Diseñe el detector de paridad con puertas lógicas, especificando la función lógica de salida, F, en función de los bits de entrada, A3 A2 A1 A0(bit menos significativo).
- 1.2. Dibuje el circuito resultante.

↓.

$b_3$	$b_2$	$b_1$	$b_0$	$P$
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	1
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	0
1	1	1	0	0
1	1	1	1	1

$$P = \bar{b}_3 \bar{b}_2 \bar{b}_1 \bar{b}_0 + \bar{b}_3 \bar{b}_2 b_1 b_0 + \bar{b}_3 b_2 \bar{b}_1 b_0 + \bar{b}_3 b_2 b_1 \bar{b}_0 + b_3 \bar{b}_2 \bar{b}_1 b_0 + b_3 \bar{b}_2 b_1 \bar{b}_0 + b_3 b_2 \bar{b}_1 \bar{b}_0 + b_3 b_2 b_1 b_0$$

$$P = \bar{b}_3 \bar{b}_2 (\bar{b}_1 \bar{b}_0 + b_1 b_0) + \bar{b}_3 b_2 (\bar{b}_1 b_0 + b_1 \bar{b}_0) + b_3 \bar{b}_2 (\bar{b}_1 b_0 + b_1 \bar{b}_0) + b_3 b_2 (\bar{b}_1 \bar{b}_0 + b_1 b_0)$$

$$\bar{b}_1 \bar{b}_0 + b_1 b_0 = \overline{b_1 \oplus b_0} \quad \bar{b}_1 b_0 + b_1 \bar{b}_0 = b_1 \oplus b_0$$

$$b_1 \oplus b_0 = m \Rightarrow$$

$$P = \bar{b}_3 \bar{b}_2 (\bar{m}) + \bar{b}_3 b_2 (m) + b_3 \bar{b}_2 (m) + b_3 b_2 (\bar{m})$$

$$P = \bar{m} (\bar{b}_3 \bar{b}_2 + b_3 b_2) + m (b_3 \bar{b}_2 + b_3 b_2)$$

$$P = \bar{m} (\overline{b_3 \oplus b_2}) + m (b_3 \oplus b_2)$$

$$P = \overline{b_1 \oplus b_2} \overline{b_3 \oplus b_2} + (b_1 \oplus b_2)(b_3 \oplus b_2) =$$

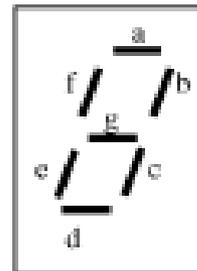
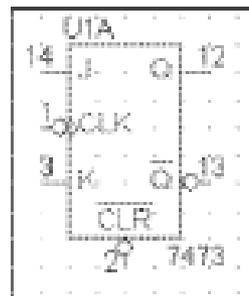
$$P = b_3 \oplus b_2 \oplus b_1 \oplus b_0$$

### 3. Diseño del decodificador de números binarios a 7-segmentos.

3.1. Diseñe con puertas un decodificador para que el número decimal equivalente al número binario de palabras cuyo número de unos es par y que va contando el contador, aparezca representado en un visualizador de 7-segmentos. El código de los segmentos es el que se muestra en la figura adjunta.

### 4. Circuito completo.

4.1. Dibuje el circuito completo.



3°

$a_3$	$a_2$	$a_1$	$a_0$	a	b	c	d	e	f	g
0	0	0	0							
0	0	0	1							
0	0	1	0							
0	0	1	1							
0	1	0	0							

Pg 197  
libro problemas