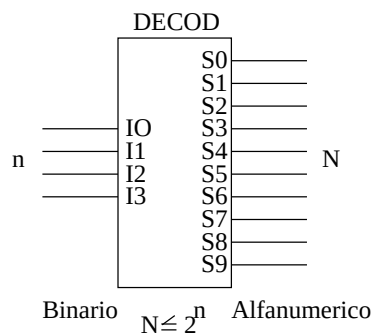


DECODIFICADOR: Convierte el codigo binario de 'n' bits de entrada , en salidas mutuamente exclusivas (decimal) o multiple salida (7 segmentos)

Decodificador BCD – decimal



I3	I2	I1	I0		S0	S1	S3 ...	S9
0	0	0	0	0	1	0	0	0
0	0	0	1	1	0	1	0	0
0	0	1	0	2	0	0	1	0
1	0	0	1	9	0	0	0	1
1	0	1	0	-	0	0	0	0
1	0	1	0	-	X	X	X	X

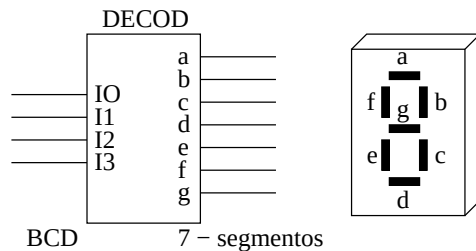
Si se completa la tabla con ceros
Desarrollo directo (No admite entradas fuera del rango BCD)

Si se completa la tabla con X
Desarrollo por Karnaugh (Admite entradas fuera del rango BCD)

$$\begin{aligned} S0 &= \overline{I3}.\overline{I2}.\overline{I1}.\overline{I0} \\ S1 &= \overline{I3}.\overline{I2}.\overline{I1}.I0 \\ S9 &= I3.I2.I1.I0 \end{aligned}$$

$$\begin{aligned} S0 &= \overline{I3}.\overline{I2}.\overline{I1}.\overline{I0} \\ S2 &= I3.I2.I1 \\ S9 &= I3.I0 \end{aligned}$$

Decodificador BCD – 7 segmentos



digito	I3	I2	I1	I0	g	f	e	d	c	b	a
0	0	0	0	0	0	1	1	1	1	1	1
1	0	0	0	1	0	0	0	0	1	1	0
2	0	0	1	0	1	0	1	1	0	1	1
9	1	0	0	1	1	1	0	1	1	1	1
- - -	1	0	1	0	x	x	x	x	x	x	x

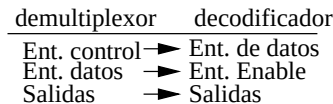
Se desarrolla cada salida (a, b,...) por Karnaugh y se obtiene como implementar el circuito

$$a = I1 + I3 + \overline{I2 \oplus I0}$$

$$c = I3 + \overline{I2} + I1$$

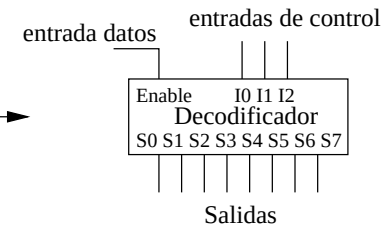
Aplicaciones con decodificadores

Realización de demultiplexores

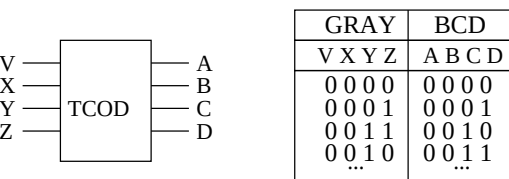


Implementacion de funciones

Expresamos la funcion como suma de productos de 'n' variables
Buscamos en la tabla del decodificador la salida correspondiente a cada producto de la funcion
Se conectan las salidas a una puerta OR (activas a nivel alto) o se conectan a una puerta NAND (activas a nivel bajo)



TRANSCODIFICADOR o CONVERSION DE CODIGO: Convierte el codigo binario de 'n' bits de entrada, en otro codigo binario de 'n' bits de salida



El circuito se obtiene como siempre. Se desarrolla por Karnaugh cada una de las salidas (A,B,C y D)

En este caso (GRAY – BCD) hay que prestar atencion a la salida, ya que se puede simplificar mas utilizando XOR