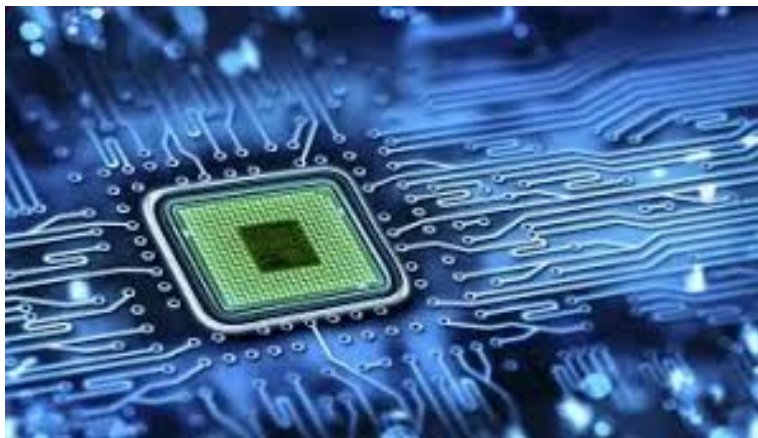


Electrónica Digital

Ingeniería Informática – FICH, UNL
Leonardo Giovanini



**Circuitos combinatoriales
para funciones lógicas**
Codificadores y decodificadores

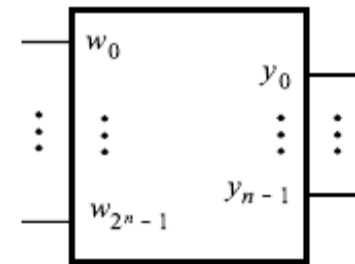
En esta se estudiarán los siguientes temas:

- Codificador;
- Codificadores binarios;
- Codificadores binarios con prioridad;
- Conversores de códigos;
- Decodificadores binarios.

Un **codificador** es un circuito lógico combinacional que **convierte símbolos de un alfabeto en otro**, es decir implementa un **proceso de codificación**.

El proceso de codificación puede implementarse por diferentes motivos, como ya vimos en la clase de códigos. Una de sus aplicaciones más comunes es **comprimir la información** para hacer más sencillo su procesamiento, almacenamiento y transmisión.

Un ejemplo típico de estos circuitos son los **codificadores binarios**, los cuales codifican la información proveniente de 2^n entradas en un código binario de n -bit. Si sólo una de las entradas puede tener el valor 1, mientras que las restantes deben permanecer en 0, las salidas identifican cual entrada es igual a 1.



A	B	f
0	0	X
0	1	0
1	0	1
1	1	X

La tabla de verdad del codificador de 2 en 1 muestra dos condiciones indefinidas (X), las cuales asumiremos que no se cumplen nunca (siempre hay solo una entrada activa). Bajo estas condiciones, su ecuación lógica es

$$f = A\bar{B}$$

La tabla de verdad muestra las entradas y salidas de un codificador de 4 en 2.

Suponiendo que estas condiciones no se cumplen nunca obtenemos los mapas de Karnaugh para cada salida del codificador

$f_1 = \overline{A}B\overline{C}D + A\overline{B}C\overline{D}$				
	C D			
AB	00	01	11	10
00	X	0	X	0
01	1	X	X	X
11	X	X	X	X
10	1	X	X	X

$f_0 = \overline{A}\overline{B}C\overline{D} + A\overline{B}C\overline{D}$				
	C D			
AB	00	01	11	10
00	X	0	X	1
01	0	X	X	X
11	X	X	X	X
10	1	X	X	X

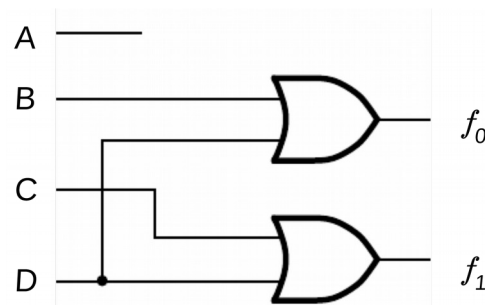
A	B	C	D	f
0	0	0	0	X
0	0	0	1	00
0	0	1	0	01
0	0	1	1	X
0	1	0	0	10
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	11
1	0	0	1	X
1	0	1	0	X
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

Un circuito simplificado que implementa la misma función está dada por

$$f_0 = B + D$$

$$f_1 = C + D$$

y su implementación está dada por



Circuitos Combinacionales – Codificadores binarios

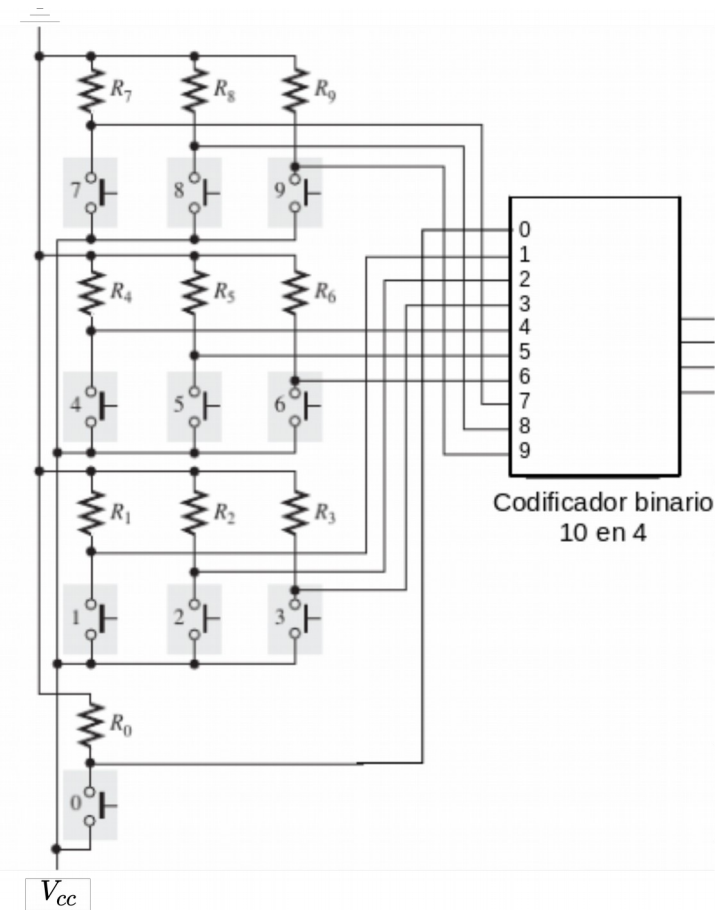
Ejemplo

Un ejemplo típico de aplicación de un codificador es un **teclado**. Se utiliza en **interfaces numéricas simples**, como alarmas, relojes y lavarropas.

El teclado considerado en este ejemplo tiene diez teclas, correspondientes a cada dígito, pero podría considerarse cualquier número de teclas.

Las teclas se implementan a través pulsadores conectados a las entradas del codificador. En este caso, las entradas son activas con unos, pero se las podría hacer activas por ceros.

Las resistencias de pull-up (resistencia conectada a masa – *Gnd* –) aseguran que todas las líneas estén a nivel lógico 0 cuando no hay tecla pulsada. Cuando se pulsa una tecla, la línea se conecta a alimentación (V_{cc}) y se aplica un nivel lógico 1 a la entrada correspondiente del codificador, generando su código binario.



Circuitos Combinacionales – Codificadores binarios con prioridad

En un codificador todas las entradas tienen la **misma importancia**. Sin embargo, hay aplicaciones en la que es necesario dar más importancia a una entrada que a las otras. Esto implica establecer un orden de prioridades a las mismas.

Si se activan **dos ó más entradas simultáneamente**, el codificador generará el código correspondiente a la **entrada de mayor prioridad**.

Esto se implementa **modificando la tabla de verdad** del codificador y **agregando una nueva salida** (z) que indique cuando se ha pulsado una tecla activa.

Estos circuitos se los conoce como **codificadores con prioridad**, y su función lógica está dada por

$f_1 = B + A$

	CD			
AB	00	01	11	10
00	X	0	0	0
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

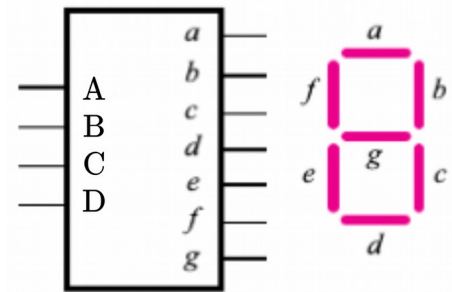
$f_0 = \overline{B}C + A$

	CD			
AB	00	01	11	10
00	X	0	1	1
01	0	0	0	0
11	1	1	1	1
10	1	1	1	1

A	B	C	D	f	z
0	0	0	0	X	1
0	0	0	1	00	0
0	0	1	0	01	0
0	0	1	1	01	0
0	1	0	0	10	0
0	1	0	1	10	0
0	1	1	0	10	0
0	1	1	1	10	0
1	0	0	0	11	0
1	0	0	1	11	0
1	0	1	0	11	0
1	0	1	1	11	0
1	1	0	0	11	0
1	1	0	1	11	0
1	1	1	0	11	0
1	1	1	1	11	0

Otra de las aplicaciones de la codificación es cambiar la representación de la información (**conversión de código**) para facilitar su procesamiento, presentación o transmisión, entre otras.

Un ejemplo típico de estos circuitos conversores de códigos son los **conversores BCD – 7 segmentos**, los cuales convierten un número binario codificado en decimal (BCD) en información adecuada para ser presentada en un display orientado a dígitos.



A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

Este circuito combinacional convierte el dígito BCD en siete señales que se utilizan para controlar los segmentos del display. Cada segmento está formado por diodo emisor de luz (LED) que emite luz cuando se lo activa con una señal eléctrica. Los segmentos están etiquetados desde *a* a *g*, y la tabla de verdad muestra la salida de cada segmento para cada valor de las entradas *A*, *B*, *C* y *D*.

Mapa de Karnaugh del segmento *a*

$$a = \overline{A}\overline{C} + \overline{A}BD + A\overline{B}\overline{D} + \overline{B}C\overline{D} \quad x=0$$

$$= A + C + BD + \overline{B}\overline{D} \quad x=1$$

	C D			
AB	0 0	0 1	1 1	1 0
0 0	1	0	1	1
0 1	0	1	1	1
1 1	X	X	X	X
1 0	1	1	X	X

Tabla de verdad

A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	X	X	X	X	X	X	X
1	0	1	1	X	X	X	X	X	X	X
1	1	0	0	X	X	X	X	X	X	X
1	1	0	1	X	X	X	X	X	X	X
1	1	1	0	X	X	X	X	X	X	X
1	1	1	1	X	X	X	X	X	X	X

¿Cómo se modifican la funciones lógicas de los segmentos si agregamos una entrada que **apague todos** los segmentos?

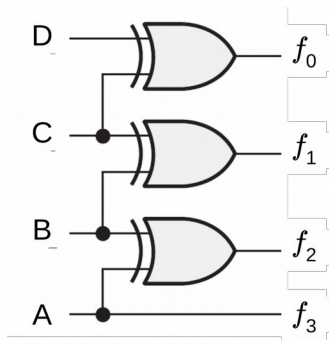
¿Cómo se modifican la funciones lógicas de los segmentos una entrada que **encienda todos** los segmentos?

Circuitos Combinacionales – Conversores de código binario a Gray

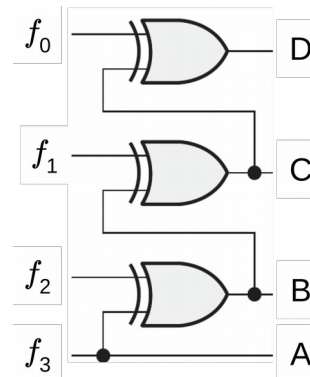
Otro ejemplo típico de estos circuitos conversores de códigos son los **conversores de binario – Gray**, los cuales convierten un número binario codificado en código Gray. La tabla de verdad establece la relación entre los cuatro dígitos del código binario con cada dígito del código Gray.

$f_3 = A$					$f_2 = \bar{A}B + A\bar{B}$					$f_1 = \bar{B}C + B\bar{C}$					$f_0 = \bar{C}D + C\bar{D}$				
AB	00	01	11	10	AB	00	01	11	10	AB	00	01	11	10	AB	00	01	11	10
00	0	0	0	0	00	0	0	0	0	00	0	0	1	1	00	0	1	0	1
01	0	0	0	0	01	1	1	1	1	01	1	1	0	0	01	0	1	0	1
11	1	1	1	1	11	0	0	0	0	11	1	1	0	0	11	0	1	0	1
10	1	1	1	1	10	1	1	1	1	10	0	0	1	1	10	0	1	0	1

Cuya implementación está dada



y su función inversa es



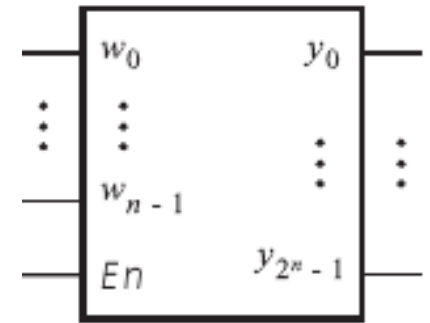
A	B	C	D	f_3	f_2	f_1	f_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

Un **decodificador** es un circuito lógico combinacional que **convierte símbolos de un alfabeto en otro**, es decir implementa la función inversa de codificador: un **proceso de decodificación**.

El proceso de decodificación puede implementarse por diferentes motivos, como ya vimos en la clase de códigos. Una de sus aplicaciones más comunes es **descomprimir la información** para hacer más sencillo su procesamiento, uso y presentación.

Un ejemplo típico de estos circuitos son los **deodificadores binarios**, los cuales **detectan la presencia de una combinación de bits** (código) en sus entradas e indica su presencia mediante una salida dada.

En su forma general, un decodificador tiene n líneas de entrada (que determina el tamaño del código) y una de las 2^n líneas de salida indica la presencia de una de las combinaciones posibles.



Los decodificadores incluyen una entrada – **En** – que **permite habilitar o deshabilitar el proceso de decodificación** en el circuito. Cuando esta entrada está activa, las salidas son 0 independiente del código presente en las entradas.

Circuitos Combinacionales – Decodificadores binarios

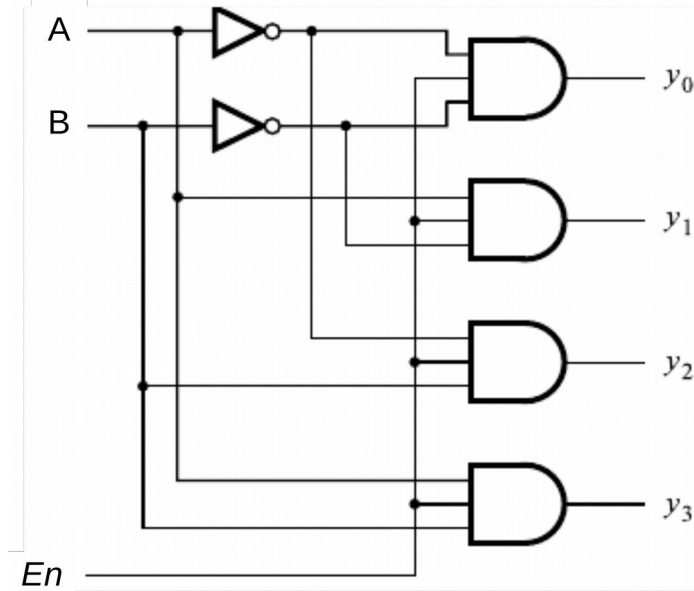
Decodificador 2 en 4

A continuación, se presenta la tabla de verdad del decodificador de 2 en 4 con señal de habilitación (En).

El decodificador se construye a partir de cuatro circuitos combinacionales que detectan cada uno de los códigos.

Se construye un mapa de Karnaugh para cada una de las salidas del decodificador (y_0 , y_1 , y_2 e y_3), de modo de obtener la función lógica correspondiente a cada una de ellas.

En	A	B	y_0	y_1	y_2	y_3
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
0	x	x	0	0	0	0



$$y_0 = A'B'En,$$

$$y_1 = A'BEn,$$

$$y_2 = AB'En,$$

$$y_3 = ABEn,$$

Notemos la función lógica de cada salida se corresponde al minitérmino del código.

Circuitos Combinacionales – Decodificadores binarios

Decodificador 4 en 16

Decodificadores con más salidas pueden construirse a partir de decodificadores más pequeños.

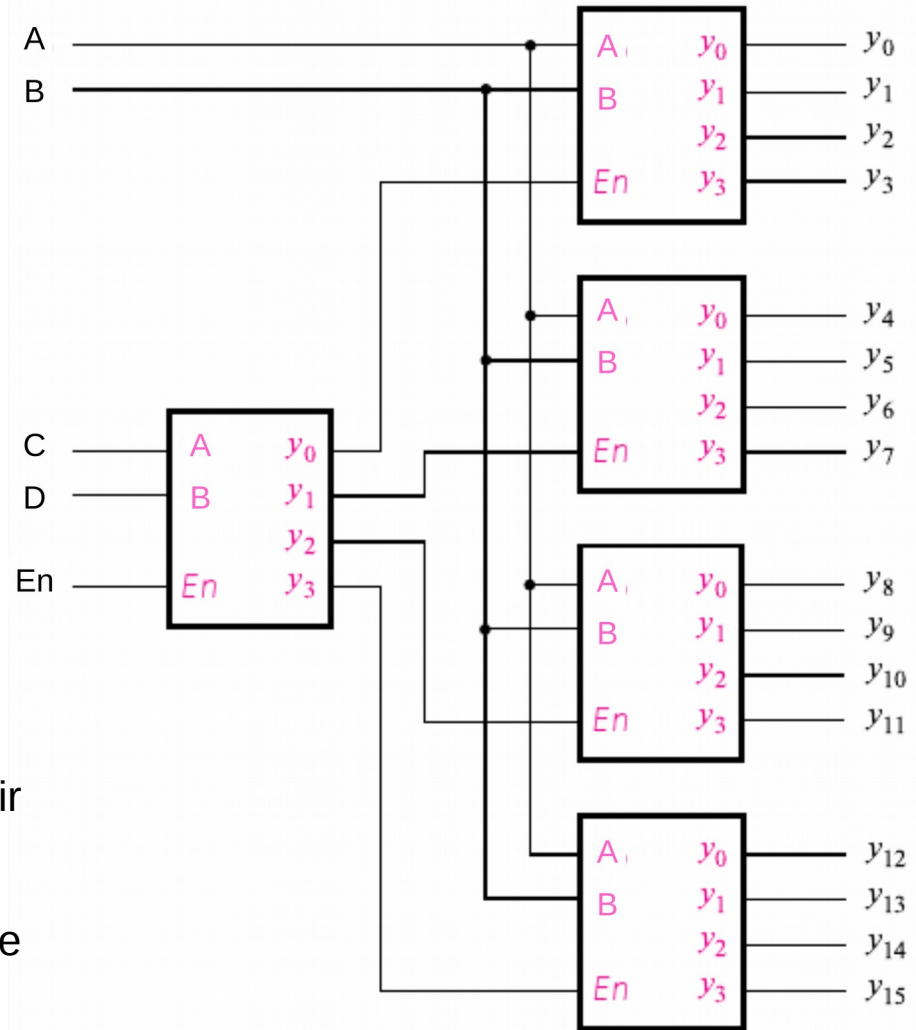
La figura muestra como se conectan cinco decodificadores 2-a-4 para construir uno 4-a-16.

Cuatro decodificadores se utilizan para generar las salidas del decodificador, están conectados a los bits bajos de la entrada.

El decodificador restante está conectado a los bits altos de la entrada y genera las señales de habilitación

Este concepto se puede utilizar para construir decodificadores de cualquier tamaño.

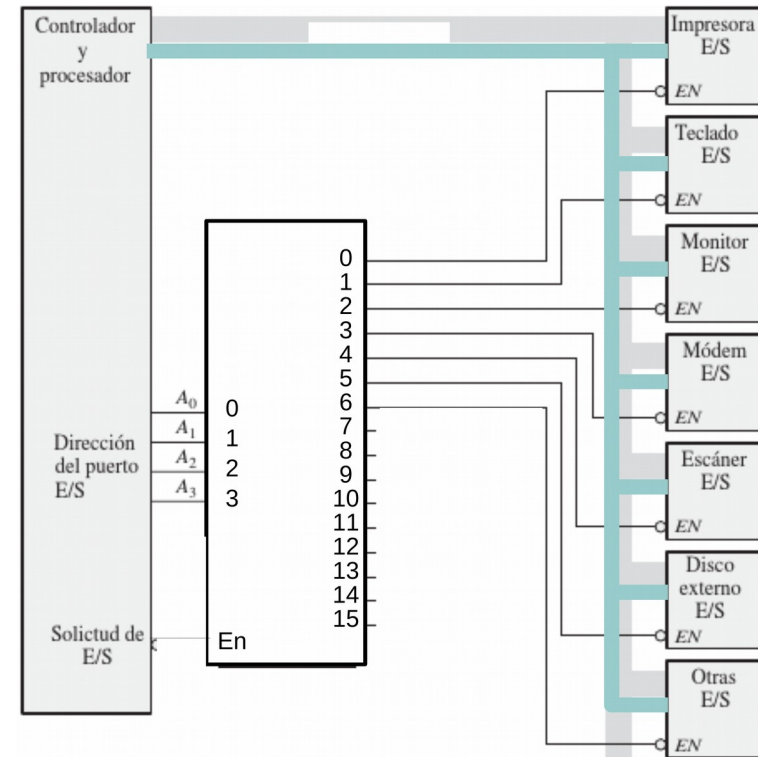
Debido a su estructura en arbol, este tipo de circuitos es como arbol de decodificación.



Un ejemplo es la selección de puertos en los microcontroladores y computadoras. Estos dispositivos externos incluyen impresoras, modems, escáneres, unidades de disco externas, teclados, monitores y otras computadoras.

Cada puerto tiene un número, denominado dirección, que lo identifica unívocamente. Cuando la computadora desea comunicarse con algún dispositivo en particular, envía la dirección del puerto al que está conectado el dispositivo en cuestión. Esta dirección del puerto se decodifica, activándose la salida del decodificador apropiada que habilita el correspondiente puerto.

Los decodificadores se utilizan para seleccionar el puerto determinado por la computadora, de forma que los datos puedan ser enviados o recibidos desde algún dispositivo concreto.



Los datos se transfieren dentro de la computadora a través de un bus de datos, que es un conjunto de líneas paralelas. El bus de datos está conectado a todos los puertos, pero los datos sólo pasarán a través del puerto que se seleccione el decodificador.

Otro ejemplo del uso de decodificadores es la selección de posiciones de almacenamiento de en una memoria de un sistema digital.

Una memoria es una colección de elementos capaces de almacenar y retener el estado de un bit. Cada una de estas posiciones tiene un número denominado ***dirección***, que lo identifica unívocamente. Cuando la computadora desea acceder a alguna de estas celdas en particular, envía la dirección al bus de direcciones. Esta dirección de la celda se decodifica, activándose la salida del decodificador apropiada que habilita la celda correspondiente.

El decodificador se utilizan para seleccionar la celda, o palabra, determinada por el sistema digital, de forma que los datos puedan ser enviados al sistema o recibidos por la celda.

