

TEMA 5. SISTEMAS COMBINACIONALES MSI.

INTRODUCCIÓN

Los sistemas combinacionales son aquellos en los que las salidas dependen exclusivamente de las entradas, luego para una misma entrada siempre se tiene la misma salida.

Hasta ahora el diseño de funciones lógicas, o de circuitos lógicos, se ha realizado mediante el uso exclusivo de puertas básicas. Los circuitos que contienen estas puertas básicas son conocidos como **SSI** (Small Scale of Integration) por que contienen un número pequeño de transistores. El diseño que se realiza con estos dispositivos se denomina **CUSTOM**.

Un paso más profundo en el diseño **HARDWARE** es realizar un diseño **SEMI-CUSTOM**, basado en el uso de bloques constructores más complejos. Esto se puede hacer mediante el uso de sistemas o circuitos **MSI** (Medium Scale of Integration) dónde el número de puertas básicas puede llegar a 100.

Más avanzados son los sistemas **LSI** (Large Scale of Integration ~1000), **VLSI** (Very Large Scale of Integration >1000), y **ULSI** (Ultra Large Scale of Integration >100000).

En un computador se realizan principalmente operaciones de codificación y decodificación de datos usando **codificadores** y **decodificadores**; transmisión y control de datos usando líneas de bus, **multiplexadores** y **demultiplexadores**; y procesamiento de datos mediante circuitería **aritmética**.

En nuestro computador podemos encontrarnos los siguientes sistemas MSI:

- CODIFICADORES Y DECODIFICADORES
- MULTIPLEXORES Y DEMULTIPLEXORES
- SUMADORES, COMPARADORES ...

Además estos dispositivos pueden usarse también para la realización de funciones complejas con un considerable ahorro de área frente al uso de puertas básicas (circuitos SSI).

DECODIFICADORES

Un decodificador es un circuito lógico con **n** entradas y **2ⁿ** salidas, tal que para cada combinación de entradas se activa al menos una salida. Si sólo se activa una salida se denomina **decodificador completo**.

Por ejemplo este es un circuito decodificador completo de 3 a 8 líneas, permitiría la activación de un dispositivo al proporcionarle la dirección de dicho dispositivo. Dispone de una entrada de **HABILITACIÓN** (enable) que conecta o desconecta el dispositivo, en este caso dicha entrada es activa a **NIVEL BAJO**, ya que el dispositivo se activa cuando dicha entrada recibe un '0' lógico.

/EN	A	B	C	D0	D1	D2	D3	D4	D5	D6	D7
1	0	0	0	X	X	X	X	X	X	X	X
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	0
0	1	0	0	0	0	0	0	1	0	0	0
0	1	0	1	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	0	0	0	1	0
0	1	1	1	0	0	0	0	0	0	0	1

Los decodificadores pueden dividirse en diferentes tipos:

- **EXCITADORES (DRIVERS)** que controlan algún dispositivo.
- **NO EXCITADORES**, los que no se usan para dicho fin.

Tanto las entradas como las salidas, principalmente estas últimas, pueden ser:

- **ACTIVAS A NIVEL ALTO**: la salida activa es **1** y la no activa **0**.
- **ACTIVAS A NIVEL BAJO**: la salida activa es **0** y la no activa **1**.

Además el número de entradas de Habilitación puede ser de una o más, y pueden estar activas a nivel alto o bajo.

Podemos encontrar decodificadores de muy diversos “tamaños”:

De 2 a 4 líneas

De 3 a 8 líneas (bin a oct)

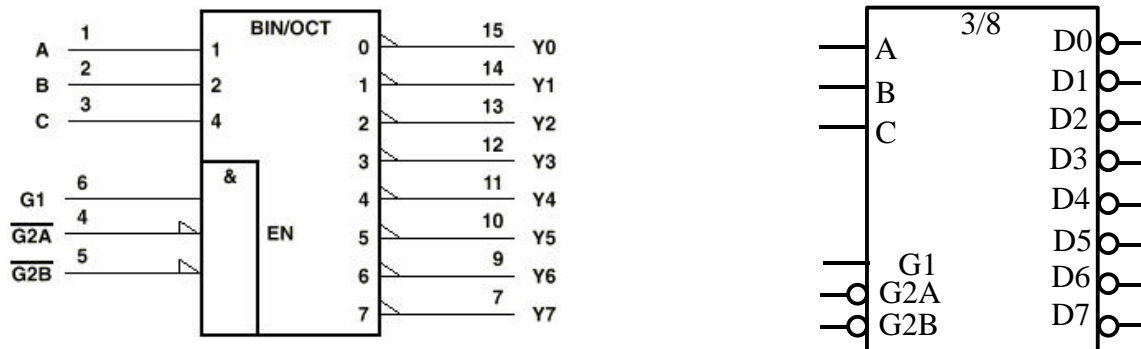
De 4 a 16 líneas (bin a hex)

Convertidores de códigos: BCD/decimal y BCD/7-seg

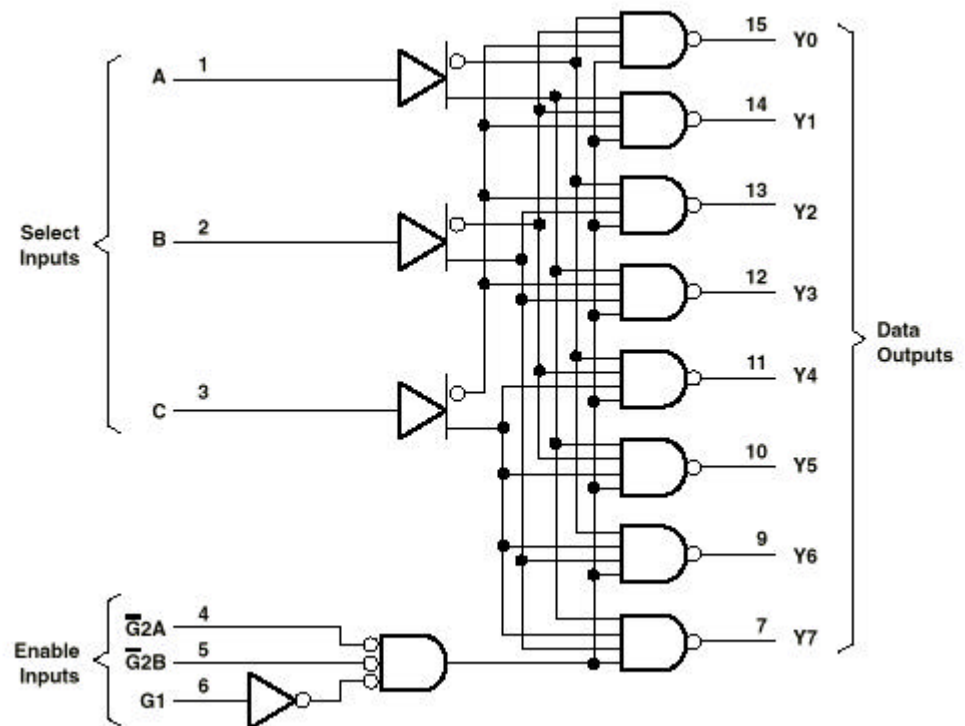
Ejemplo de Decodificador completo de 3 a 8 líneas: CIRCUITO 74X138

FUNCTION TABLE													
INPUTS						OUTPUTS							
ENABLE			SELECT										
G1	$\overline{G2A}$	$\overline{G2B}$	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	X	H	H	H	H	H	H	H	H
X	X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	L	H	H	H	H	L	H	H	H	H	H
H	L	L	H	L	L	H	H	H	H	L	H	H	H
H	L	L	H	L	H	H	H	H	H	H	L	H	H
H	L	L	H	H	L	H	H	H	H	H	H	L	H
H	L	L	H	H	H	H	H	H	H	H	H	H	L

Simbolos Lógico del Decodificador, según el Standard IEEE y tradicional.



Interior del decodificador:

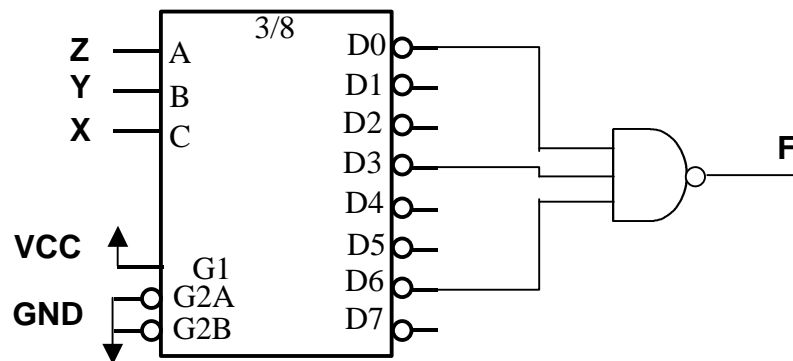


REALIZACIÓN DE FUNCIONES CON DECODIFICADORES

Un circuito decodificador completo genera todos los productos fundamentales (minterminos) de las variables de entrada.

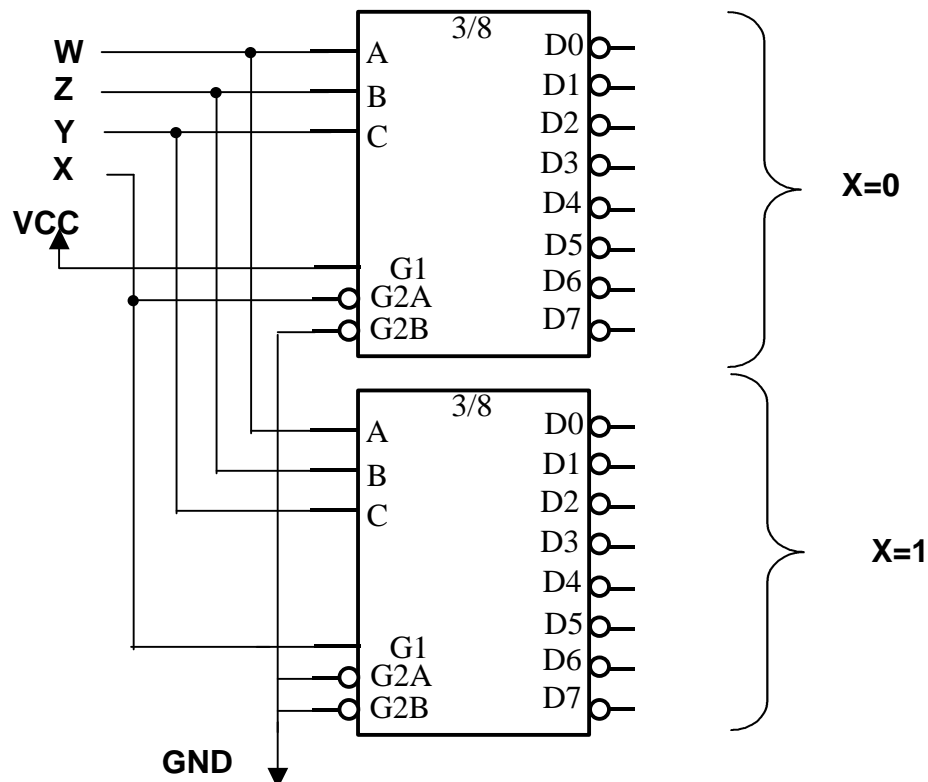
Cuándo las salidas del decodificador son activas a **nivel bajo**, para realizar la función en suma de productos basta con conectar las salidas correspondientes a los minterminos de la función usando puertas NAND:

Por ejemplo: $F(X,Y,Z) = \sum m(0, 3, 6)$



A veces puede ocurrir que necesitemos decodificar más líneas de las que nos permite nuestro circuito, se debe entonces construir un decodificador de mayor tamaño usando decodificadores de menor tamaño:

Por ejemplo para 4 bits (X,Y,Z,W)

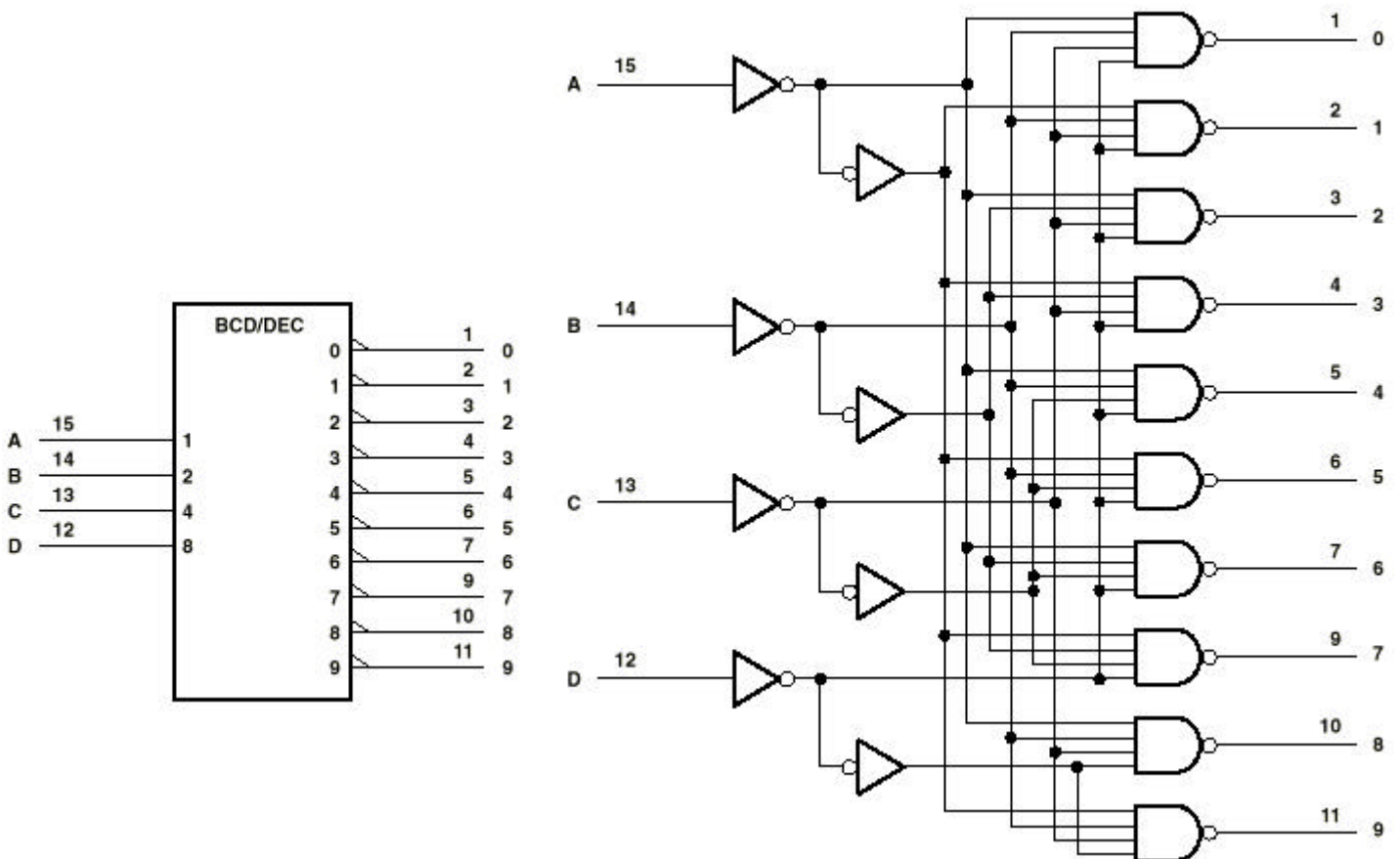


CONVERTIDORES DE CÓDIGOS

Son circuitos Codificadores/Decodificadores que convierten los datos de un código a otro.

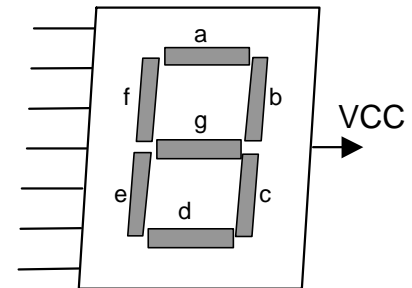
El circuito 74X42 es un convertidor de BCD a Decimal (de 4 a 10 líneas).

FUNCTION TABLE														
NO.	INPUTS				OUTPUTS									
	D	C	B	A	0	1	2	3	4	5	6	7	8	9
0	L	L	L	L	L	H	H	H	H	H	H	H	H	H
1	L	L	L	H	H	L	H	H	H	H	H	H	H	H
2	L	L	H	L	H	H	L	H	H	H	H	H	H	H
3	L	L	H	H	H	H	H	L	H	H	H	H	H	H
4	L	H	L	L	H	H	H	H	L	H	H	H	H	H
5	L	H	L	H	H	H	H	H	H	L	H	H	H	H
6	L	H	H	L	H	H	H	H	H	H	L	H	H	H
7	L	H	H	H	H	H	H	H	H	H	H	L	H	H
8	H	L	L	L	H	H	H	H	H	H	H	H	L	H
9	H	L	L	H	H	H	H	H	H	H	H	H	H	L
Invalid	H	L	H	L	H	H	H	H	H	H	H	H	H	H
	H	L	H	H	H	H	H	H	H	H	H	H	H	H
	H	H	L	L	H	H	H	H	H	H	H	H	H	H
	H	H	L	H	H	H	H	H	H	H	H	H	H	H
	H	H	H	L	H	H	H	H	H	H	H	H	H	H



El circuito 74X47 es un convertidor de BCD a 7-segmentos (de 4 a 7 líneas). Permite visualizar en un display de 7-segmentos el dígito decimal correspondiente al BCD. Son circuitos drivers.

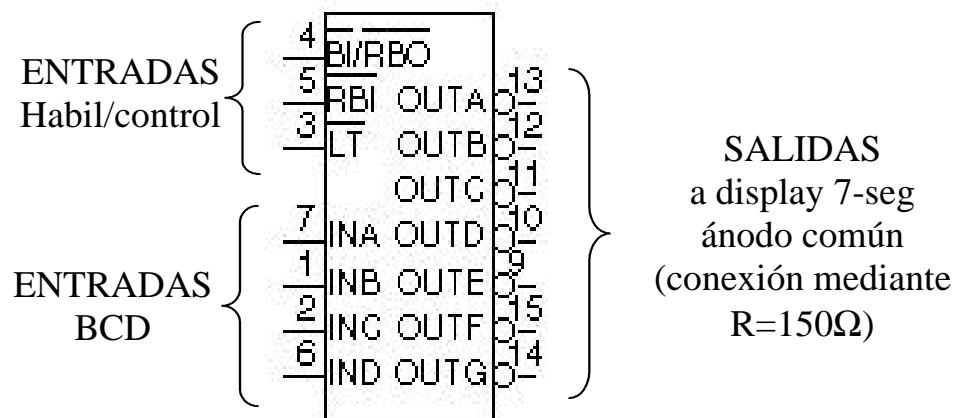
Los visualizadores están formados por siete LEDs (Light Emitter Diode), que son dispositivos que emiten luz cuando la corriente que los atraviesa excede de un cierto valor (al igual que un diodo normal conduce cuando se supera una determinada tensión entre sus bornes), de ahí que para su excitación se necesiten dispositivos que proporcionen corriente suficiente. Existen dos tipos de visualizadores con LEDs principalmente. Los de **ánodo común** y los de **cátodo común**. Los primeros se usan cuando el decodificador tiene salidas activas a nivel bajo, mientras que los segundos son para los decodificadores con salidas activas a nivel alto.



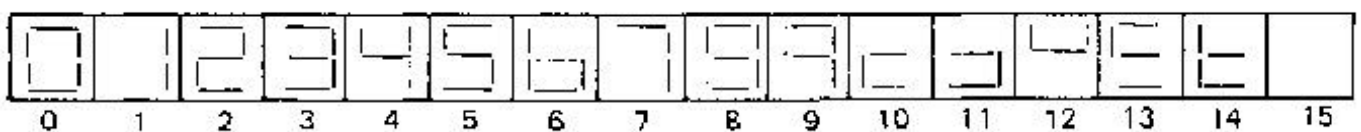
ÁNODO COMÚN

Otros visualizadores muy usados son los LCD (Liquid Crystal Display) que no están formados por diodos sino por una serie de plaquitas conductoras capaces de excitar un líquido que hay entre ellas.

Símbolo lógico y explicación de los terminales de I/O:



Asignación numérica de las entradas y resultado de la visualización



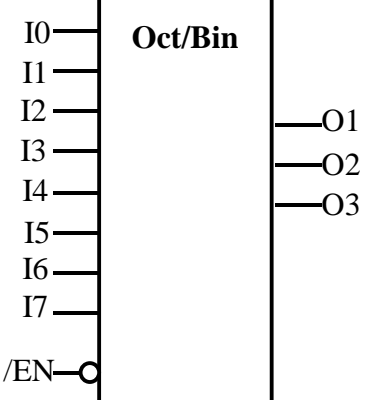
CODIFICADORES

Son los dispositivos MSI que realizan la operación inversa a la realizada por los decodificadores. Generalmente, poseen 2^n entradas y n salidas.

Cuando solo una de las entradas está activa para cada combinación de salida, se le denomina **codificador completo**.

Por ejemplo, el siguiente circuito proporciona a la salida la combinación binaria de la entrada que se encuentra activada. En este caso se trata de un codificador completo de 8 bits, o también llamado codificador de 8 a 3 líneas:

/EN	I0	I1	I2	I3	I4	I5	I6	I7	O1	O2	O3
1	X	X	X	X	X	X	X	X	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	0	1	1	1	1



Las salidas codificadas, generalmente se usan para controlar un conjunto de 2^n dispositivos, suponiendo claro está que sólo uno de ellos está activo en cualquier momento. Sin embargo cuando nos encontremos con que se deben controlar dispositivos que pueden estar activos al mismo tiempo, problema que se suelen encontrar los sistemas microprocesadores, es preciso usar un dispositivo que nos proporcione a la salida el código del dispositivo que tenga más alta prioridad.

En la siguiente página podemos ver una sencilla comparación.

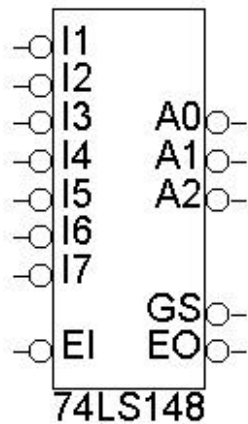
En la siguiente figura se representa el diagrama lógico de un codificador completo de Decimal a BCD natural, junto a su tabla de funcionamiento.

I1	I2	I3	I4	I5	I6	I7	I8	I9	A	B	C	D
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1
0	0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	0	0	1	0	0	0	0	1	1	0
0	0	0	0	0	0	1	0	0	0	1	1	1
0	0	0	0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	1

Por otro lado la figura siguiente representa el diagrama lógico del circuito 74147, que es un codificador de **prioridad** de Decimal a BCD natural; en la tabla de funcionamiento adjunta se puede notar la diferencia con el anterior.

/I1	/I2	/I3	/I4	/I5	/I6	/I7	/I8	/I9	/A	/B	/C	/D
X	X	X	X	X	X	X	X	0	0	1	1	0
X	X	X	X	X	X	X	0	1	0	1	1	1
X	X	X	X	X	X	0	1	1	1	0	0	0
X	X	X	X	X	0	1	1	1	1	0	0	1
X	X	X	X	0	1	1	1	1	1	0	1	0
X	X	X	0	1	1	1	1	1	1	0	1	1
X	X	0	1	1	1	1	1	1	1	1	0	0
X	0	1	1	1	1	1	1	1	1	1	0	1
0	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

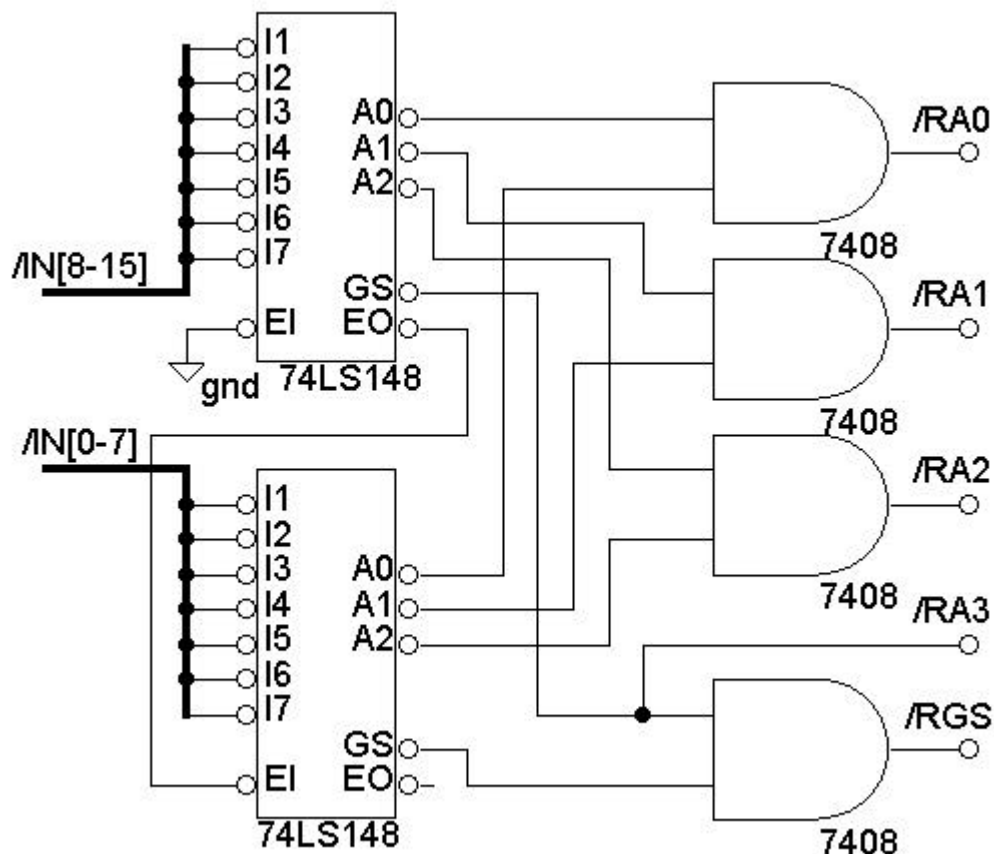
Cuando se trata de establecer la prioridad con mayor número de bits, es preciso recurrir a la asociación de codificadores. El siguiente diagrama muestra un codificador de prioridad de 16 líneas a 4, usando codificadores de prioridad 74148, de 8 a 3 líneas.



/EI: Habilitación

/GS: es 0 cuando el dispositivo está habilitado y una o más de sus entradas está activa

/EO: salida para habilitar otro codificador de más baja prioridad



MULTIPLEXORES

Multiplexar es pasar información de “muchos” canales o líneas a “pocos” canales o líneas.

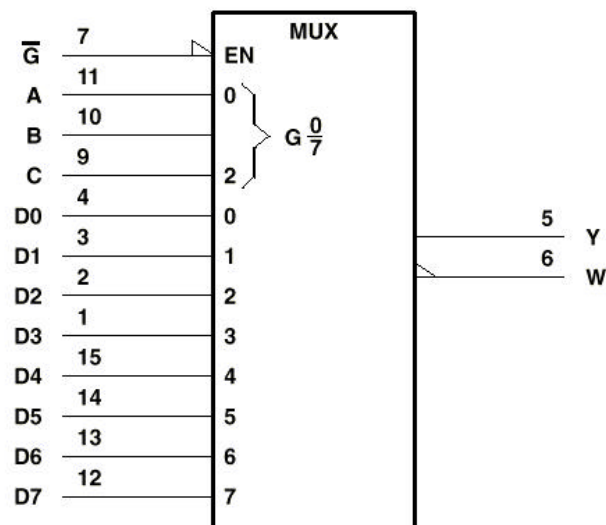
Un **MULTIPLEXOR (MUX)** es un circuito combinacional que selecciona una entrada y la transfiere a la salida. La selección de la entrada, o dato, se realiza según un conjunto de valores de las variables de control.

Poseen por tanto, **n** entradas de selección, para **2ⁿ** entrada de datos, proporcionando **dos salidas**: una para el dato directo y otra para el dato negado.

A continuación se presenta la tabla de funcionamiento y el Símbolo lógico estándar para un multiplexor de 8 a 1 líneas. Se trata del circuito 74X151, con entrada de habilitación activa a nivel bajo.

FUNCTION TABLE					
INPUTS				OUTPUTS	
SELECT			STROBE \overline{G}		
C	B	A		Y	W
X	X	X	H	L	H
L	L	L	L	D0	D0
L	L	H	L	D1	$\overline{D1}$
L	H	L	L	D2	$\overline{D2}$
L	H	H	L	D3	$\overline{D3}$
H	L	L	L	D4	$\overline{D4}$
H	L	H	L	D5	$\overline{D5}$
H	H	L	L	D6	$\overline{D6}$
H	H	H	L	D7	$\overline{D7}$

H = high level, L = low level, X = irrelevant
D0, D1, . . . D7 = the level of the respective D input



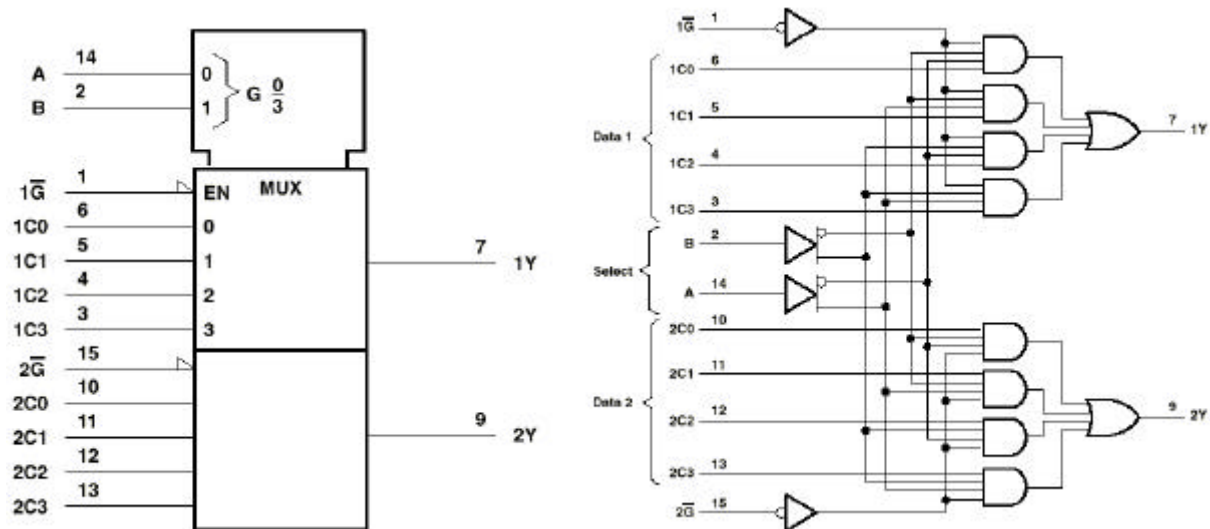
Existen en el mercado diferentes multiplexores.

El siguiente corresponde al 74153, que es un circuito MSI con dos Mux de 4 a 1 líneas.

A los Mux se les suele llamar también selectores de datos. Así este sería un selector de datos 1-de-4 (ya que selecciona un dato de cuatro disponibles).

FUNCTION TABLE							
INPUTS						STROBE \overline{G}	OUTPUT Y
SELECT		DATA					
B	A	C0	C1	C2	C3		
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
L	H	X	L	X	X	L	L
L	H	X	H	X	X	L	H
H	L	X	X	L	X	L	L
H	L	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H

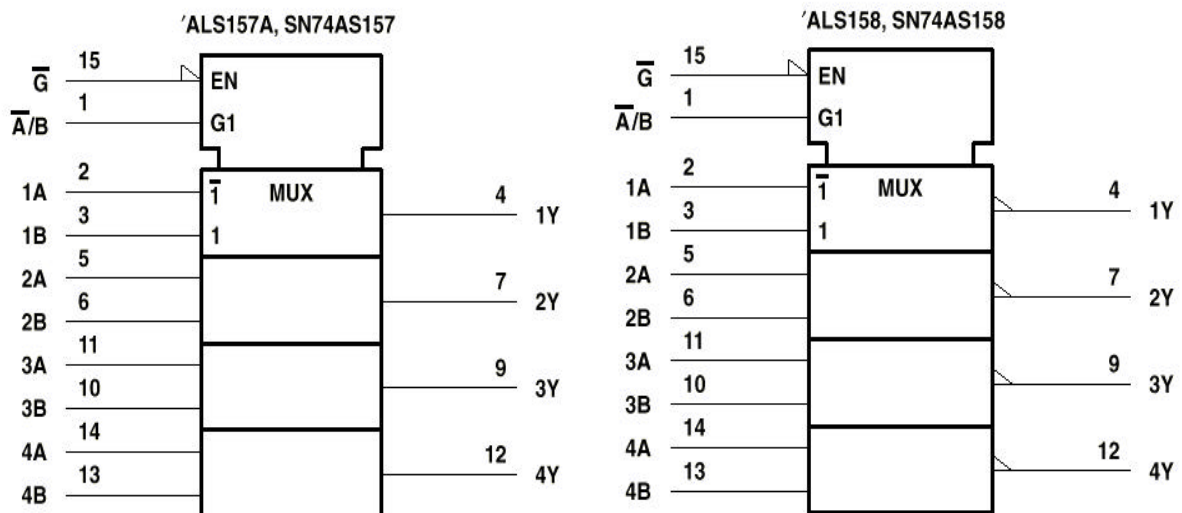
Select inputs A and B are common to both sections.



El diagrama lógico sobre estas líneas corresponde a la configuración interior del dispositivo arriba descrito, lo que nos puede dar una idea de la diferencia entre los circuitos SSI y los MSI.

Un multiplexor muy usado es el 74157 (El 74158 es una versión del primero con las salidas activas a nivel bajo). Se trata de un circuito con cuádruple multiplexor de dos a una líneas:

FUNCTION TABLE					
INPUTS				OUTPUT Y	
\overline{G}	$\overline{A/B}$	DATA		'ALS157A SN74AS157	'ALS158 SN74AS158
		A	B		
H	X	X	X	L	H
L	L	L	X	L	H
L	L	H	X	H	L
L	H	X	L	L	H
L	H	X	H	H	L



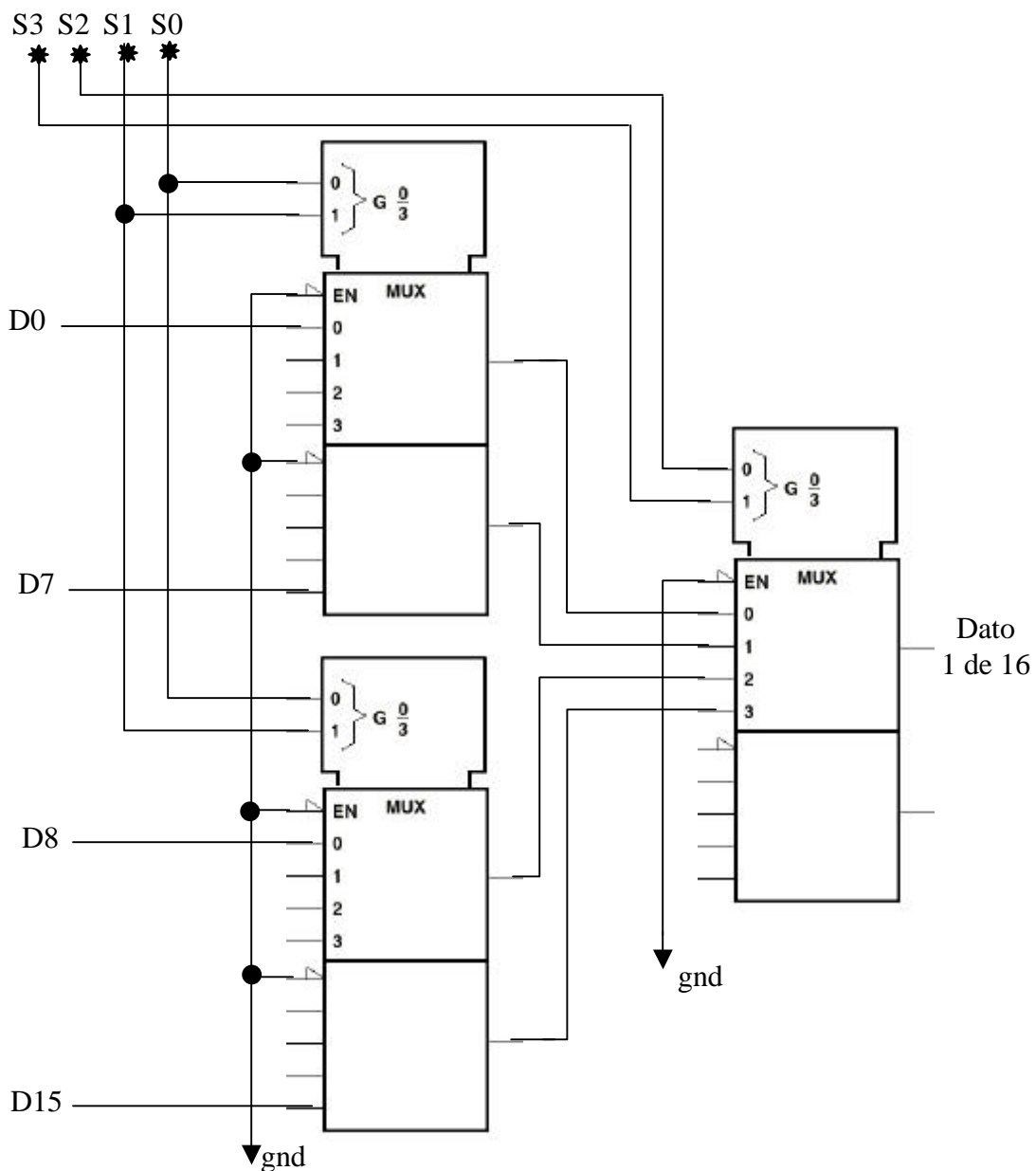
Este Mux funciona como selector de palabras: según sea el valor de selección $G1$, en las cuatro salidas aparece $A[1-4]$ ó $B[1-4]$, lo que lo hace muy útil en las aplicaciones aritméticas que se verán en el tema siguiente.

EXTENSIÓN DE MULTIPLEXORES

Como siempre, cuando no se dispone del número necesario de entradas en un solo dispositivo, hay que recurrir a asociar dispositivos de menor número de entradas para obtener uno mayor.

Vamos a construir un MUX de 16:1 usando MUX de 4:1.

La asociación es muy similar a la que se hace con los decodificadores, pero cambian la disposición de los MSB y LSB (bits más significativos y menos significativos).



REALIZACIÓN DE FUNCIONES COMBINACIONALES CON MUX

Partimos de la siguiente afirmación: un multiplexor de 2^n entradas puede realizar cualquier función lógica de $n+1$ variables.

Se pueden usar dos métodos:

- de forma algebraica;
- de forma tabular.

Nos centraremos en este último, y lo seguiremos mediante un ejemplo:

1. A partir de la expresión canónica y se escoge un Mux determinado:

Ej: Sea $f(A,B,C,D) = \sum_4 m(0,2,3,7,8,13,15)$

Al ser una función de 4 variables necesitamos un MUX de 8 a 1 líneas (o sea, con tres variables de control)

2. Se crea un mapa de Karnaugh de manera que la numeración en las columnas, coincida con la entrada que se pretende seleccionar.

Así, las columnas, vendrán determinadas por las variables de control del MUX, y las filas por el dato o los datos que se quieren transmitir. Las variables de control deben ser las de menor peso. Evaluando cada columna identificamos el valor que hay que colocar en cada entrada.

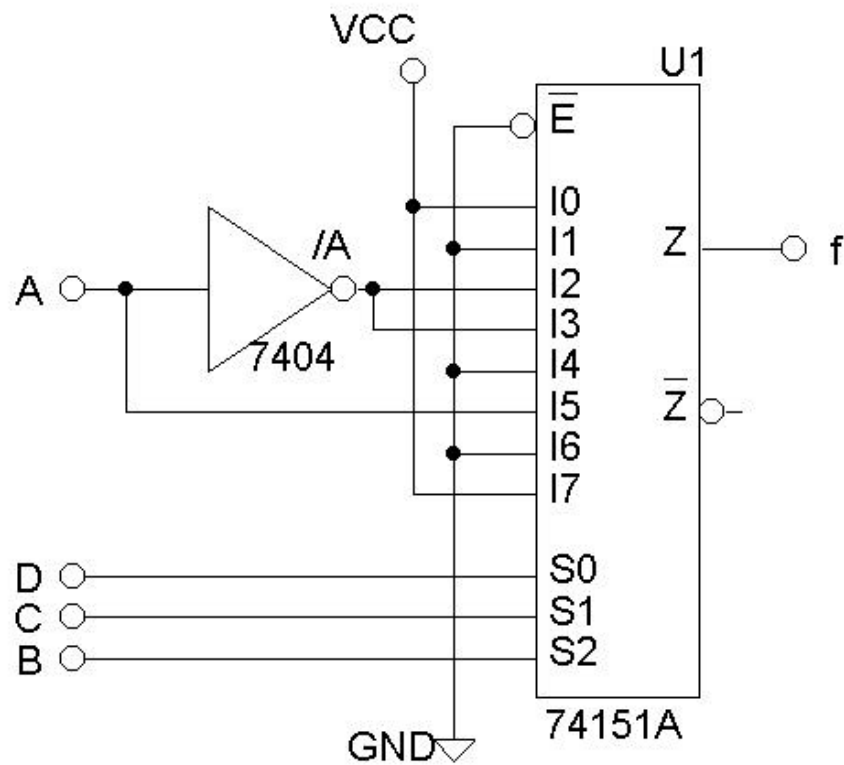
Ej: Realizamos dicho mapa para nuestra función:

		BCD							
		I0	I1	I2	I3	I4	I5	I6	I7
		000	001	010	011	100	101	110	111
A	0	1 0	0 1	1 2	1 3	0 4	0 5	0 6	1 7
	1	1 8	0 9	0 10	0 11	0 12	1 13	0 14	1 15
		1	0	/A	/A	0	A	0	1

3. Hacemos el diagrama lógico del circuito colocando en las entradas de datos lo que la tabla nos indique.

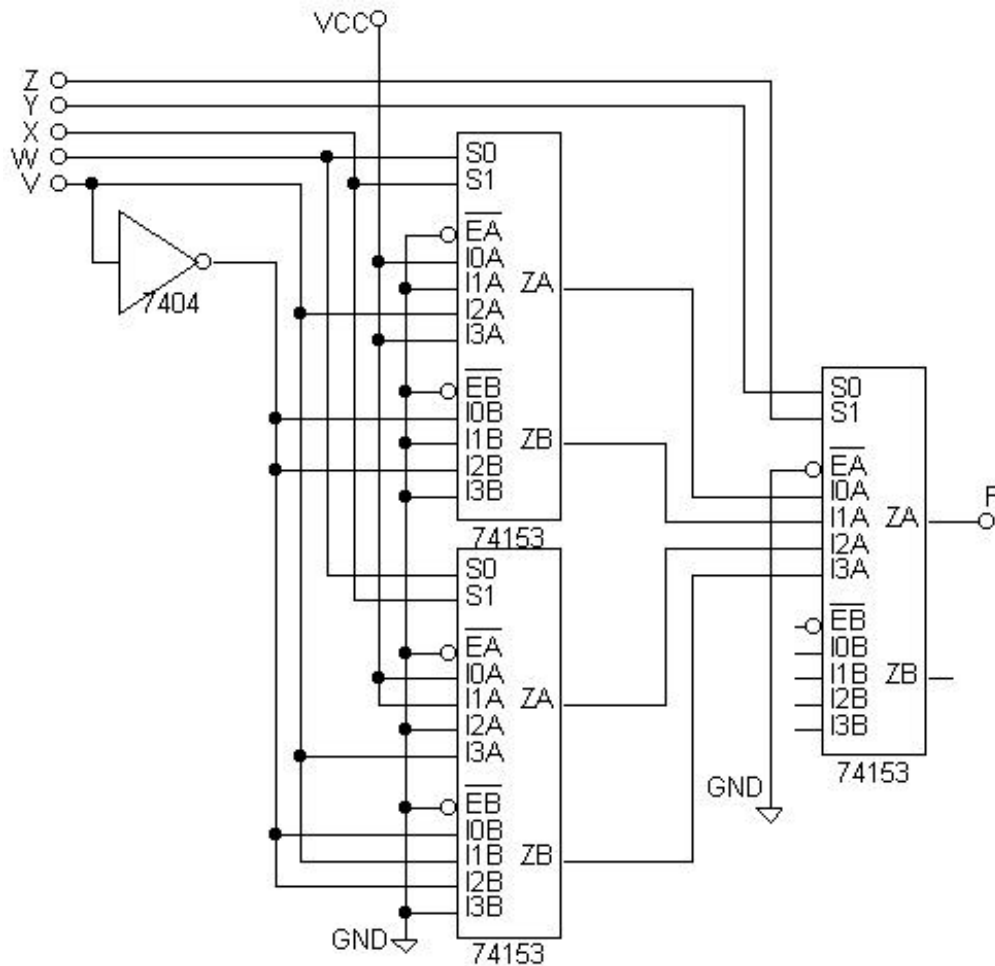
Ej: Nuestra tabla nos dice que en la entrada **I0** de nuestro MUX, debemos colocar un **1**; que en la **I1**, un **0**; ...

Nos debe quedar un circuito como el de la figura:



EJERCICIO DE MUX

Para familiarizarnos con el empleo de MUX, deduzcamos la función que realiza el siguiente circuito



DEMULTIPLEXORES

En realidad no existen como tales, sino que vienen definidos por los decodificadores/demultiplexores.

La función que debe realizar es la **inversa** de la que realiza el MUX, o sea, debemos seleccionar una salida por donde transmitir el dato de la entrada.

Por tanto, el circuito constará de **1** entrada de datos, **n** entradas de selección de salida, y **2ⁿ** salidas.

El Decodificador/DEMUX 74138 que ya conocemos utiliza su entrada de habilitación **G1** para entrada de Datos:

FUNCTION TABLE													
INPUTS						OUTPUTS							
ENABLE			SELECT										
G1	$\overline{G2A}$	$\overline{G2B}$	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	X	H	H	H	H	H	H	H	H
X	X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	L	H	H	H	H	H	L	H	H	H	H
H	L	L	H	L	L	H	H	H	H	L	H	H	H
H	L	L	H	L	H	H	H	H	H	H	L	H	H
H	L	L	H	H	L	H	H	H	H	H	H	L	H
H	L	L	H	H	H	H	H	H	H	H	H	H	L

