

# MULTIPLEXORES

**GRUPO:** 7

**ALUMNOS :** JAVIER PÉREZ GUTIÉRREZ.

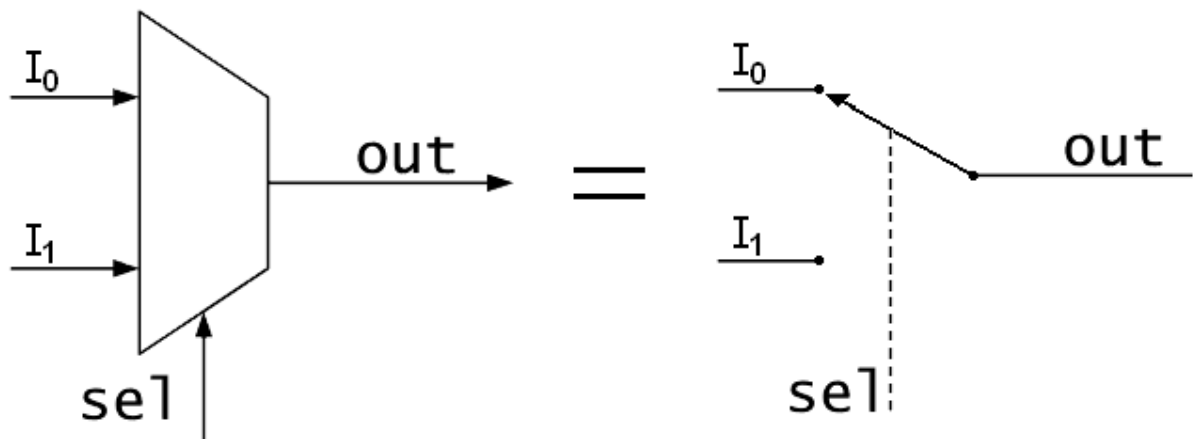
SAMUEL ECHEVRRÍA ACEREDA.

PABLO COSÍO.

## Definición:

Un multiplexor es un circuito digital que selecciona una de entre varias entradas de datos  $I_i$  y lleva su valor lógico a la única salida  $Z$  del circuito. La selección de los datos se realiza mediante una o varias entradas de control  $S_j$ . La codificación binaria resultante de las entradas  $S$  indica el índice de la entrada  $I$  que pasa a la salida. Existiendo una entrada de habilitación (enable), la cual pone en funcionamiento el circuito, y trabaja en bajo activo.

## ESQUEMA BÁSICO DEL FUNCIONAMIENTO DE UN MULTIPLEXOR 2 INPUT-MUX

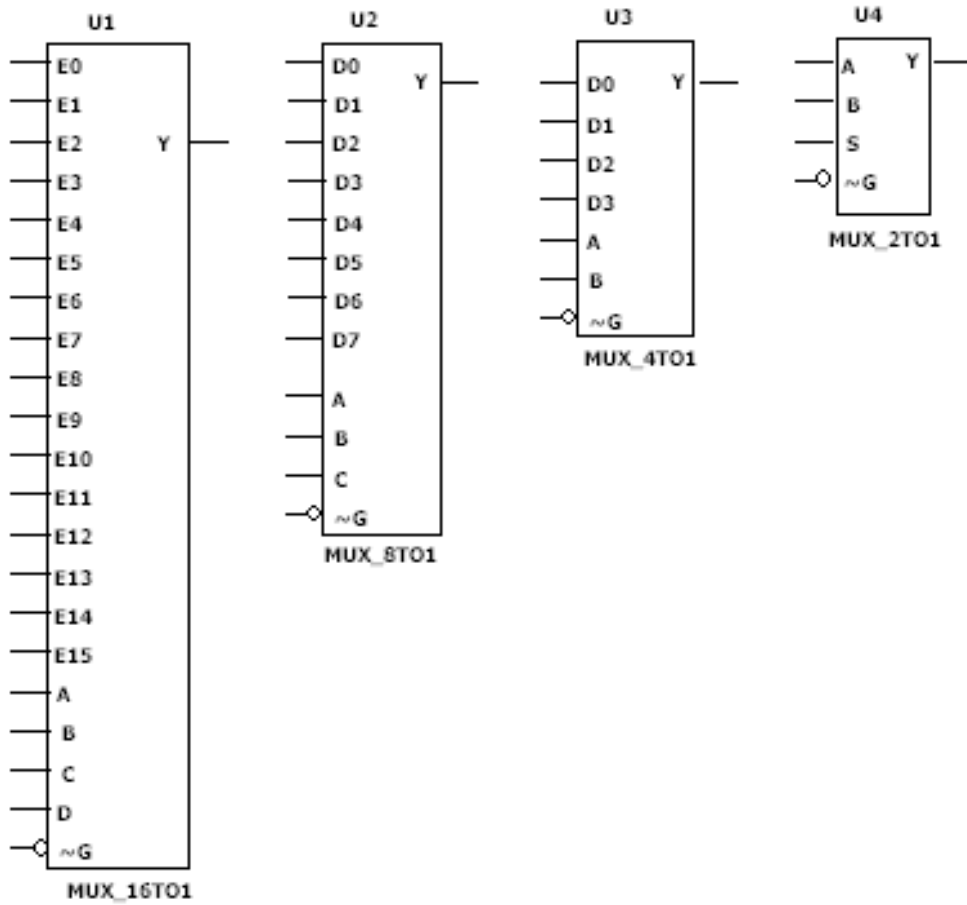


\*\*\*En el esquema se puede apreciar que existen dos entradas posibles, y el valor que tenga la salida depende de la señal de control SEL.

## ESQUEMA DE MULTIPLEXORES

Su funcionamiento es como sigue:

Con la combinación binaria presente en las entradas de selección, escogemos la entrada de información cuyo valor se va a transmitir a la salida.



Diseño un multiplexor de 2 canales de entrada.

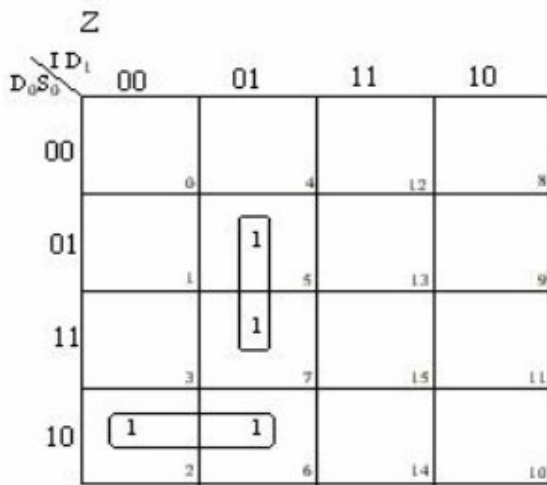
1º) Se procede a la implementación de una tabla de verdad, en la cual se aprecian los valores de la salida dependiendo de el valor de la señal de control y de la señal de habilitación(enable):

Ejemplos de explicación:

- 1-.Si el Enable está en alto activo,el circuito está apagado.
- 2-.Si el Enable está en bajo activo,el circuito funciona y selecciona la entrada atribuida a la señal de control.

Enable	D <sub>1</sub>	D <sub>0</sub>	S <sub>0</sub>	Z
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	X	X	X	0

2º) Minimización mediante mapas de Karnaugh:

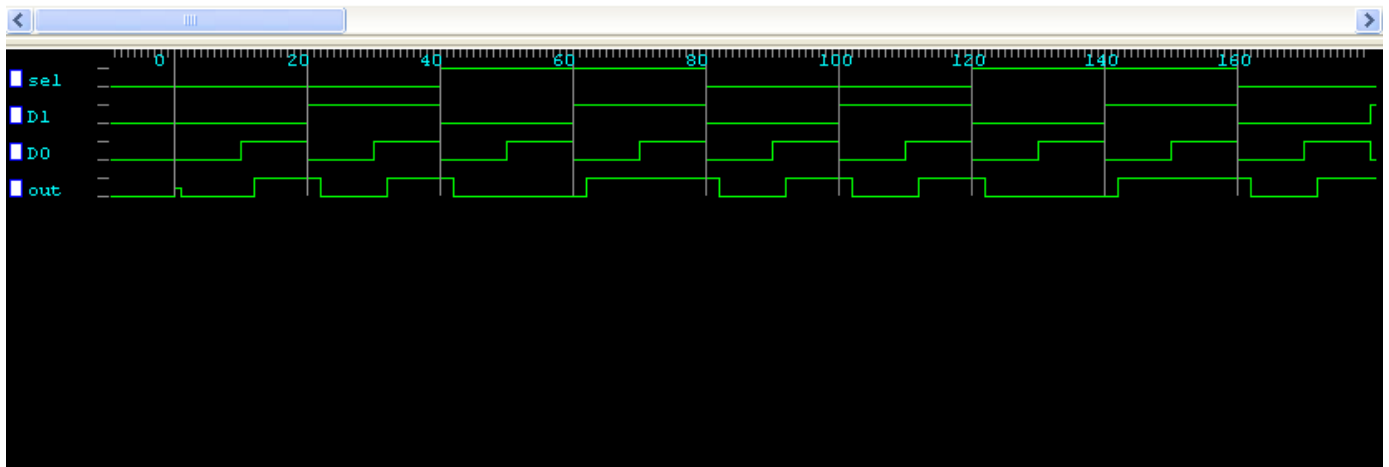
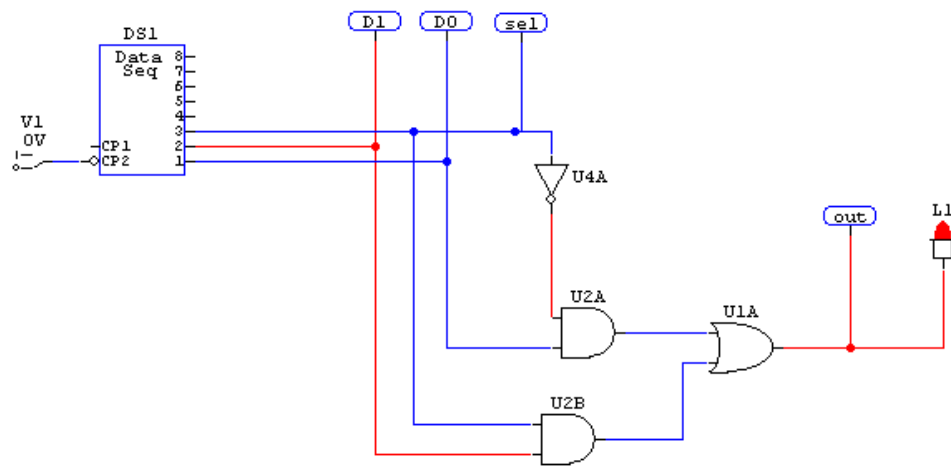


**Función en forma SOP:**

$$\sum(2,5,6,7)$$

$$Z = \bar{1}D_1 S_0 + \bar{1}D_0 \bar{S}_0$$

### 3º) Diseño del circuito y simulación mediante puertas lógicas.



#### 4º) Programación de un multiplexor en lenguaje VHDL

```
library ieee;  
use ieee.std_logic_1164.all;  
entity multiplexor4to1 is  
port (I: in std_logic_vector(3 downto 0); ← Entradas de datos  
      S: in std_logic_vector(1 downto 0); ← Entradas de selección  
      E: in std_logic; ← Entrada de habilitación  
      Z: out std_logic); ← Salida  
end mux4;
```

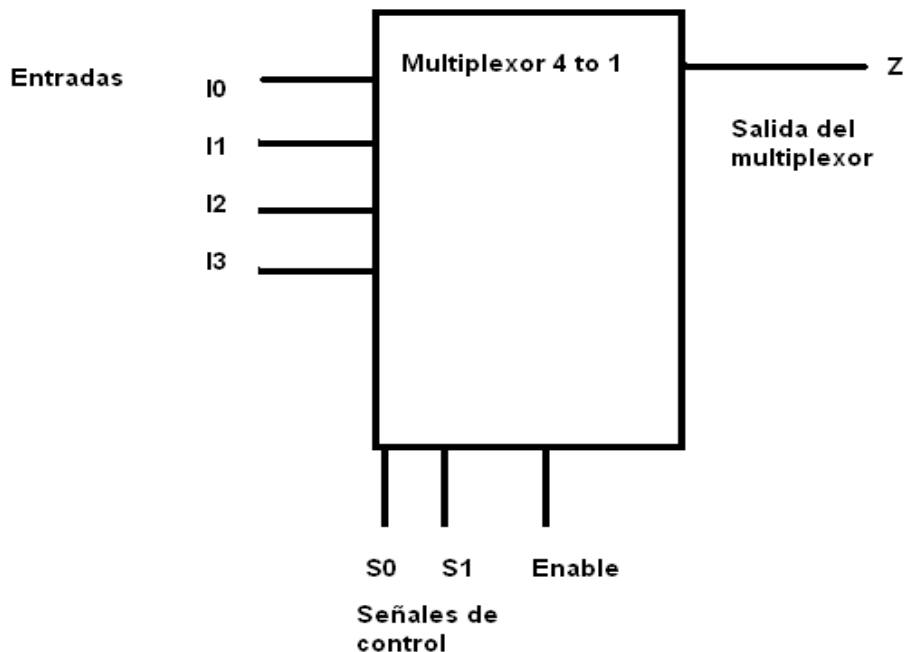
```
architecture multiplexor of multiplexor4to1 is  
begin  
process (I, S, E)  
begin
```

```
    if E = '1' then  
        Z <= '0';
```

← Si el enable está desactivado, la entrada directamente es 0  
\*\*El enable debe ser bajo activo

```
    else  
    case S is  
        when "00" => Z <= I(0);  
        when "01" => Z <= I(1);  
        when "10" => Z <= I(2);  
        when "11" => Z <= I(3);  
        when others => Z <= I(3);
```

```
    end case;  
    end if;  
    end process;  
end multiplexor;
```



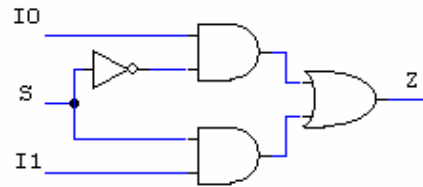
## OBTENCIÓN DE UN MULTIPLEXOR MEDIANTE PUERTAS LÓGICAS

Si se desea contruir un multiplexor, con las características que se quiera, se puede hacer un diseño mediante tablas de verdad y a continuación, implementarlo mediante puertas lógicas:

### 2-INPUT MUX

S	Z
0	I0
1	I1

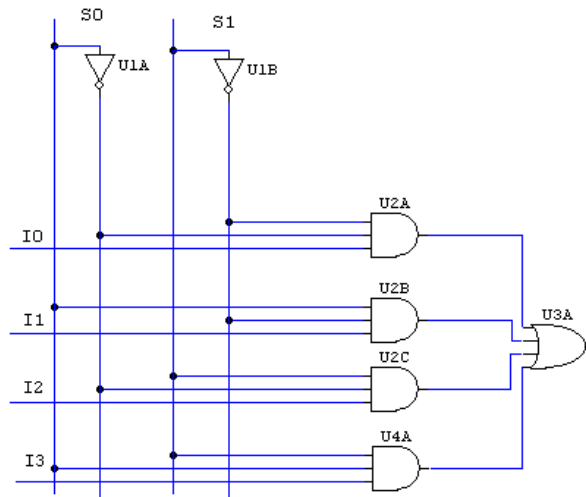
$$Z = \overline{S}I0 + SI1$$



### 4-INPUT MUX

S0	S1	Z
0	0	I0
0	1	I1
1	0	I2
1	1	I3

$$Z = \overline{S1}\overline{S0}I0 + S1\overline{S0}I1 + \overline{S1}S0I2 + S1S0I3$$



### Problema:

Dada la siguiente tabla de verdad, construir un multiplexor de 8 canales implementado puertas lógicas:

### 8-INPUT MUX

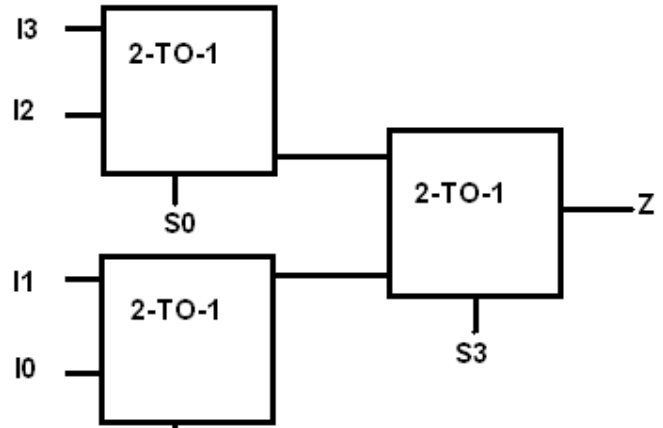
S0	S1	S2	Z
0	0	0	I0
0	0	1	I1
0	1	0	I2
0	1	1	I3
1	0	0	I4
1	0	1	I5
1	1	0	I6
1	1	1	I7

## OBTENCIÓN DE UN MULTIPLEXOR MEDIANTE COMBINACIONES DE MULTIPLEXORES

Contrucción de un multiplexor de 4 canales combinando dos multiplexores de dos canales:

La explicación es la siguiente:  
Cada uno de los multiplexores 2-to-1 selecciona una de las dos entradas disponibles dependiendo del valor de la señal de control.

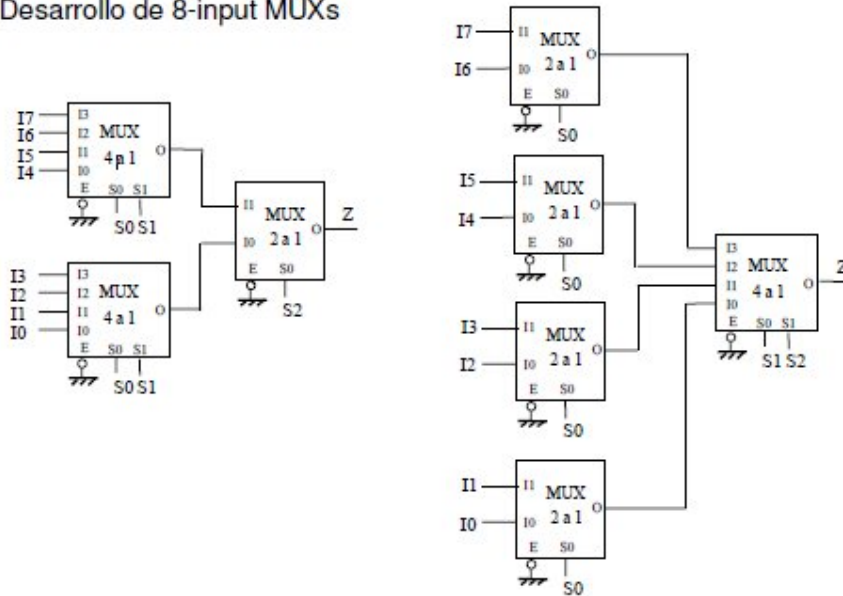
$$Z = \overline{S_3} * (\overline{S_1} * I_0 + S_1 * I_1) + S_3 * (\overline{S_0} * I_2 + S_0 * I_3)$$



### Multiplexores

Desarrollo de N-input MUXs en base a M-input MUX (N > M)

Desarrollo de 8-input MUXs



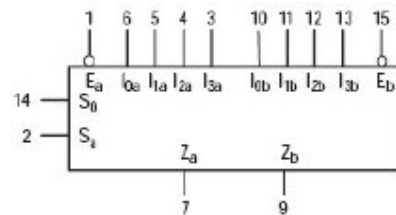
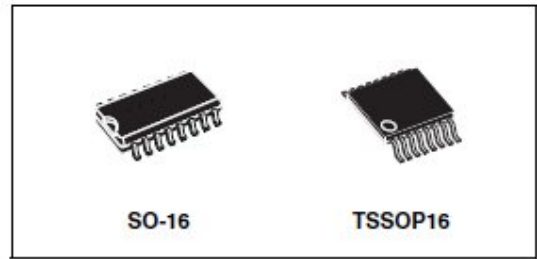
\*\*Desarrollo de un mutiplexor de 8 canales mediante combinaciones de multiplexores de 4-to-1 y 2-to-1.

## MULTIPLEXORES COMERCIALES

### MODELO: 74LCX157      2 channel multiplexer

#### Features

- 5V tolerant inputs
- High speed:
  - $t_{PD} = 6.0ns$  (Max) at  $V_{CC} = 3V$
- Power down protection on inputs and outputs
- Symmetrical output impedance:
  - $|I_{OH}| = I_{OL} = 24mA$  (Min) at  $V_{CC} = 3V$
- PCI bus levels guaranteed at 24mA
- Balanced propagation delays:
  - $t_{PLH} \approx t_{PHL}$
- Operating voltage range:
  - $V_{CC}$  (Opr) = 2.0V to 3.6V
- Pin and function compatible with 74 series 157
- Latch-up performance exceeds 500mA (JESD 17)
- ESD performance:
  - HBM > 2000V
  - (MIL STD 883 method 3015); MM > 200V



#### Truth table

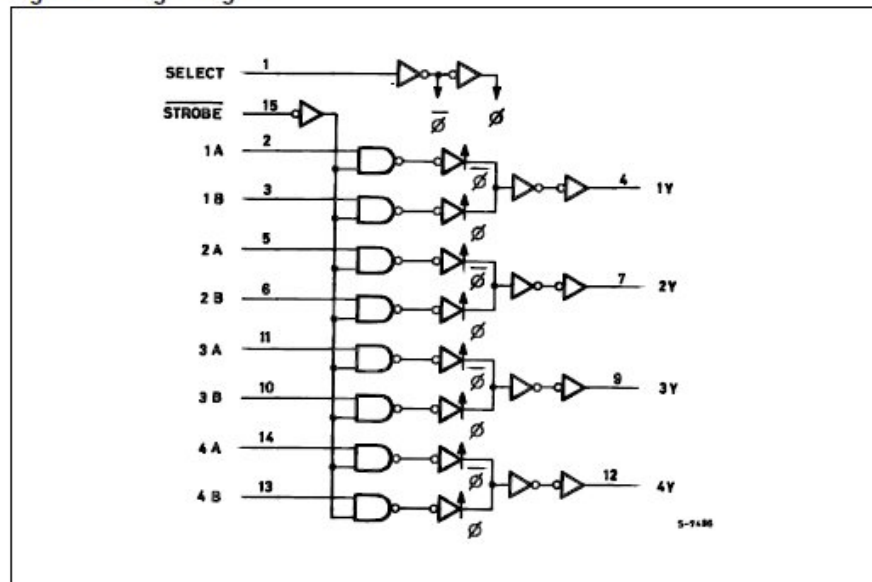
Table 2. Truth table

Inputs				Output
STROBE	SELECT	A	B	Y
H	X	X	X	L
L	L	L	X	L
L	L	H	X	H
L	H	X	L	L
L	H	X	H	H

*X : Do not care*

#### Logic diagram

Figure 4. Logic diagram





## **PROBLEMAS:**

### **1.-Diseño de un multiplexor de 5 canales:**

Combinando multiplexores como hemos visto anteriormente, podemos fácilmente construir un multiplexor de 5 canales.

