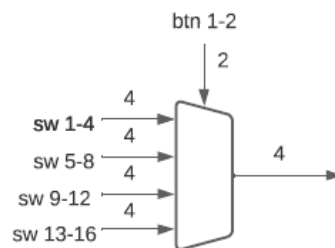


## Ej. 3: Decodificador para display 7 segmentos

### Marco teórico

#### Multiplexor

El uso de un multiplexor consiste fundamentalmente en la selección de una de varias entradas en un dispositivo. Haciendo uso de un puerto de control de selección se determina con valor binario el número de la entrada que ha de ser la salida del dispositivo. Para el caso en cuestión, el bloque de un multiplexor puede apreciarse en la Figura

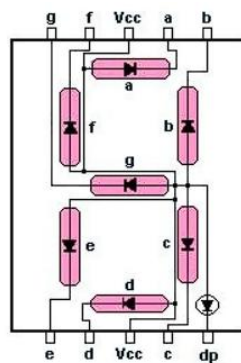


#### Decodificador de 7 segmentos BCD

Un decodificador de 7 segmentos BCD convencional consiste en un dispositivo lógico que recibe un valor binario en sus entradas y da un resultado combinacional correspondiente a la misma, esta salida ha de corresponder a al número de la entrada para que pueda ser interpretado por un display de 7 segmentos.

#### Display de 7 segmentos.

Es un dispositivo electrónico que se utiliza convencionalmente para mostrar números haciendo uso de luces led. Existen 2 tipos, de ánodo común y de cátodo común. Cada uno con una lógica opuesta para su funcionamiento, uno funciona con ceros lógicos para encender una luz y el otro con unos lógicos para encender una luz. Un ejemplo de un display de 7 segmentos se puede apreciar en la Figura



## Tabla de la verdad para un decodificador de 7 segmentos

Teniendo en cuenta el funcionamiento de un decodificador y display de 7 segmentos se puede realizar la siguiente tabla de la verdad con el fin apreciar el funcionamiento esperado principalmente por parte del decodificador BCD a 7 segmentos

N	Entradas					Salidas						
	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		1	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

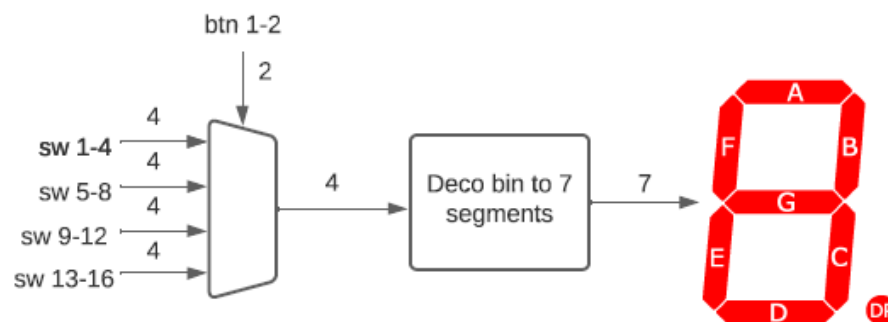
## Lista de requerimientos

Este ejercicio solicita que cumpla con los siguientes requerimientos

- 4 entradas de 4 bits denominadas *switches*
- Etapa de selección de una de las 4 entradas (multiplexaje)
- 1 entrada de control de 2 bits para la etapa de multiplexaje
- Etapa de decodificado del número seleccionado a código BCD
- Salida correspondiente al código BCD esperado
- Implementación de un testbench para pruebas
- Implementable para FPGA

## Diagrama de bloques

El diagrama de bloques para este ejercicio se puede apreciar a continuación.



# Descripción del código a implementar

## Módulos 1: TOP

Este módulo contiene las entradas principales del sistema, el único cable que posee es el conecta la salida del multiplexor y la entrada del decodificador. En este módulo también se hacen las instancias del resto de módulos para luego ser probadas en la testbench. El módulo se puede apreciar a continuación.

```
module TOP (  
    input  logic [3:0] input_mux_1,  
    input  logic [3:0] input_mux_2,  
    input  logic [3:0] input_mux_3,  
    input  logic [3:0] input_mux_4,  
    input  logic [1:0] btn_Control,  
    output logic [6:0] Result  
);  
    wire [3:0] mux_deco;
```

Cada una de las entradas corresponden a las entradas del multiplexor junto a su línea de control, mientras que su salida corresponde al resultado proveniente del decodificador.

## Módulos 2: MUX

Este módulo lleva a cabo la operación de multiplexaje. Al hacer uso de una máquina de estados implementada por medio de la función case, se determina a partir del input btn cuál de las entradas llegará a la salida. Este módulo corresponde al siguiente.

```
module MUX (  
    input  logic [3:0] sw14,  
    input  logic [3:0] sw58,  
    input  logic [3:0] sw912,  
    input  logic [3:0] sw1316,  
    input      [1:0] btn,  
    output logic [3:0] out_mux  
);
```

## Módulos 3: DECO

Este módulo sigue una lógica parecida a la del MUX, utiliza una maquina de estados para pasar a la salida llamada “Deco\_out” un valor binario específico en función del valor de “input\_deco” la cual a su vez corresponde a la salida del multiplexor. Este módulo se puede apreciar a continuación.

```
module DECO (  
    input  logic [3:0] input_deco,  
    output logic [6:0] Deco_out  
);
```

## Funcionamiento

En la sección de Testbench se hace la introducción de los valores para probar. Un correcto funcionamiento consiste en ingresar 4 números distintos en cada una de las entradas del módulo TOP y por medio de la entrada de selección del multiplexor seleccionar alguno de ellos. Para probar este funcionamiento se introducen los siguientes números para pruebas.

```
input_mux_1 = 4'd3;  
input_mux_2 = 4'd2;  
input_mux_3 = 4'd7;  
input_mux_4 = 4'd9;  
  
#40;  
btn_Control = 2'b01;
```

En este caso se introdujeron los valores de 3,2,7 y 9 respectivamente en las entradas 1,2,3 y 4 del multiplexor. La entrada de control debe cumplir la función correspondiente al “case” del módulo MUX.

```
always_comb begin  
  case(btn)  
    2'b00: out_mux = sw14;  
    2'b01: out_mux = sw58;  
    2'b10: out_mux = sw912;  
    2'b11: out_mux = sw1316;  
    default: out_mux = 4'b0000;  
  endcase  
end
```

En función del valor introducido en btn se escoge el valor de la entrada que pasa a la salida del MUX. En el caso del caso mostrado se eligió la combinación 01, correspondiente al segundo estado de la máquina de estados del MUX, es decir, la entrada sw58, es decir el número 2 previamente ingresado. Esta salida pasa por el cable “mux\_deco” al momento de hacer la instancia en el módulo Top llega a la entrada del decodificador llamada “input\_deco”.

Una vez en el módulo de DECO, la máquina de estados de cumplir las instrucciones mostradas en la Tabla 1. Por último, se corre el programa y se obtiene la salida.

```
# -----  
# El numero seleccionado corresponde a: 1101101  
# -----  
# quit -f
```

El cual corresponde al número 2 para el display de 7 segmentos.