

## INGENIERÍA DE COMPUTADORES 3

### Trabajo Práctico - Junio 2013

#### INSTRUCCIONES

- El trabajo práctico debe realizarse de manera individual. No puede realizarse en grupo. Por ello, se penalizará cualquier uso compartido de las soluciones propuestas y de los códigos programados.
- El trabajo debe entregarse a través del curso virtual de la asignatura en la plataforma Alf.
- La fecha límite de entrega es el día 16 de abril.
- El alumno debe entregar un fichero comprimido, en formato zip o tar, que contenga:
  - Una memoria en la cual explique la solución a los ejercicios, incluyendo los listados documentados del código VHDL desarrollado. Este documento deberá estar en formato pdf.
  - Los ficheros del código VHDL solución a los ejercicios.

El nombre del fichero comprimido debe ser la concatenación del nombre y apellidos del alumno. Por ejemplo, LuisaGomezMartin.zip

## CRITERIOS DE EVALUACIÓN DEL TRABAJO

- Para que el trabajo pueda ser corregido, es imprescindible que el alumno entregue dentro del plazo establecido un fichero comprimido que contenga tanto la memoria en formato pdf, como el código VHDL de los ejercicios que haya realizado.
- El trabajo se compone de 2 ejercicios con varios apartados. En el enunciado se indica la puntuación de cada apartado.
- Para aprobar el trabajo es necesario que la suma de la nota obtenida en los dos ejercicios sea igual o mayor que 5.
- Si el código VHDL solución de un apartado tiene uno o varios errores de compilación, o no tiene la funcionalidad pedida, dicho apartado se valorará con cero puntos.
- Si el código solución de un apartado compila sin errores y tiene la funcionalidad pedida, la puntuación en dicho apartado será al menos el 80 % de la nota del apartado.
- Se valorará positivamente la adecuada documentación del código, así como la presentación y calidad de las explicaciones proporcionadas en la memoria.

## PROCEDIMIENTO DE EVALUACIÓN EN ESTA ASIGNATURA

Para aprobar la asignatura es necesario aprobar el trabajo y aprobar el examen.

Plantearémos un trabajo para la convocatoria ordinaria (junio) y otro diferente para la convocatoria extraordinaria (septiembre). Este trabajo que está leyendo corresponde a la convocatoria ordinaria de 2013.

La nota obtenida en la convocatoria ordinaria en el trabajo y en el examen se guarda para la convocatoria extraordinaria. Es decir:

- Si un alumno aprueba el trabajo de la convocatoria ordinaria y no aprueba el examen, se le guarda la nota del trabajo para la convocatoria extraordinaria. Es decir, no debe hacer el trabajo de la convocatoria extraordinaria.
- Si un alumno no entrega o suspende el trabajo en convocatoria ordinaria, pero sí aprueba el examen en convocatoria ordinaria, entonces se le guarda la nota del examen para la convocatoria extraordinaria, debiendo aprobar el trabajo de la convocatoria extraordinaria para superar la asignatura.

La nota del trabajo y del examen no se guarda de un curso para otro.

## EJERCICIO 1

En la Figura 1.1 se muestra el símbolo lógico de un circuito digital cuya función es encender una luz de aviso de un coche. Este circuito enciende la luz de aviso cuando se cumplen simultáneamente las tres condiciones siguientes: la llave del coche está puesta en el contacto, existe un pasajero en el asiento del copiloto y no está abrochado el cinturón del copiloto.

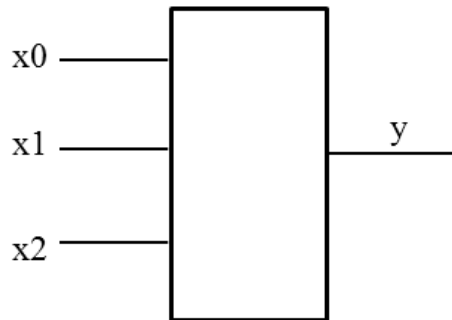


Figura 1.1: Entradas y salida del circuito de luz de aviso.

La señal  $x_0$  está a 1 sólo si la llave del coche está en el contacto,  $x_1$  está a 1 sólo si existe un pasajero en el asiento del copiloto y  $x_3$  está a 1 sólo si el cinturón del asiento del copiloto está abrochado. La señal de salida  $y$  está a 1 sólo cuando la luz de aviso está encendida.

- 1.a) (0.5 puntos) Escriba en VHDL la **entity** del circuito digital de alarma empleando el mismo nombre para las señales que el mostrado en la Figura 1.1.
- 1.b) (1 punto) Escriba la tabla de la verdad del circuito digital de alarma. A partir de dicha tabla de la verdad, obtenga la función lógica que describe la salida ( $y$ ) en función de las entradas ( $x_0, x_1, x_2$ ). A continuación, escriba en VHDL una **architecture** que describa el *comportamiento* de un circuito que implemente dicha función lógica.
- 1.c) (0.5 puntos) Dibuje el diagrama al nivel de puertas lógicas de un circuito que implemente esta función. Emplee para ello puertas lógicas AND de dos entradas y NOT. A continuación, escriba en VHDL la **entity** y la **architecture** de cada una de las puertas lógicas que componen el diagrama que acaba de dibujar.
- 1.d) (1 punto) Escriba en VHDL una **architecture** que describa la *estructura* del circuito que ha dibujado, instanciando y conectando las puertas lógicas que ha diseñado anteriormente.

- 1.e) (1 punto) Escriba en VHDL un banco de pruebas que permita visualizar, para todos los posibles valores de las entradas, la salida de los circuitos diseñados en los Apartados 1.b y 1.d. Compruebe mediante inspección visual que los dos diseños funcionan correctamente.

## EJERCICIO 2

En la Figura 1.2 se muestra el símbolo lógico de un circuito combinacional desplazador junto con su tabla de operaciones. Obsérvese que este circuito tiene una entrada de selección  $s$  de dos bits, que determina la operación que realiza el circuito sobre la señal de entrada  $x$  para obtener la señal de salida  $y$ .

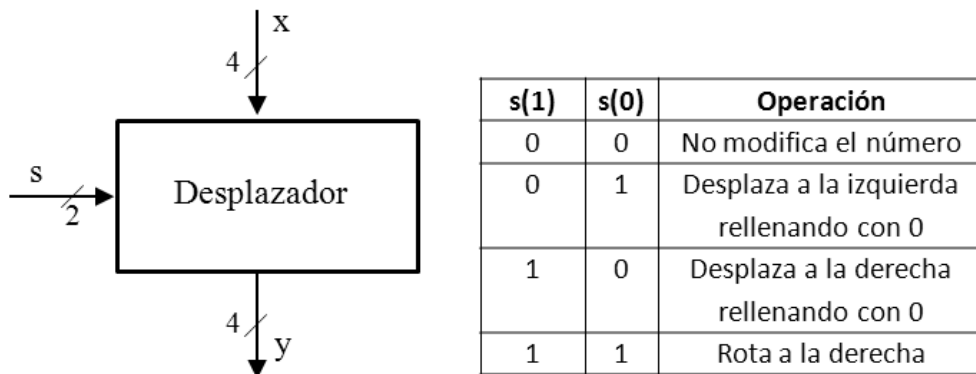


Figura 1.2: Entradas y salidas, y tabla de operaciones del circuito desplazador.

La **entity** del circuito desplazador se muestra a continuación.

```
entity shifter is
    port ( y : out std_logic_vector(3 downto 0);
          x : in  std_logic_vector(3 downto 0);
          s : in  std_logic_vector(1 downto 0) );
end entity shifter;
```

- 2.a) (2 puntos) Diseñe en VHDL la **architecture** que describe el comportamiento del circuito combinacional desplazador.
- 2.b) (1 punto) Diseñe en VHDL la **architecture** que describe el comportamiento de un multiplexor 4 a 1.

La **entity** del multiplexor ha de ser la siguiente:

```

entity mux is
  port ( y  : out std_logic;
        x  : in  std_logic_vector(3 downto 0);
        s  : in  std_logic_vector(1 downto 0) );
end entity mux;

```

- 2.c)** (1 punto) Diseñe en VHDL la **architecture** que describe la estructura del circuito combinacional desplazador, la cual debe estar compuesta mediante la conexión de multiplexores 4 a 1 como el diseñado al resolver el Ejercicio 2.b.
- 2.d)** (2 puntos) Programe en VHDL un banco de pruebas que testee todas las posibles entradas al circuito desplazador. El banco de pruebas debe comparar las salidas de la UUT con las salidas esperadas, mostrando el correspondiente mensaje de error en caso de que las salidas obtenidas de la UUT no correspondan con las esperadas. Emplee este banco de pruebas para comprobar los diseños realizados al contestar a los Apartados 2.a y 2.c.