

INSTRUCCIONES:

1. Resuelva este ejercicio en las mismas condiciones en que realizará el examen: dos horas de tiempo y sin emplear ningún material.
2. Revise sus contestaciones, empleando para ello el texto y el simulador que esté usando para estudiar la asignatura.
3. Compare sus respuestas revisadas con la solución.

Pregunta 1 (1.5 puntos)

Dibuje el diagrama conceptual correspondiente a:

1.a) (0.75 puntos) Las sentencias **if** anidadas siguientes:

```
if boolean_expr_1 then
  if boolean_expr_2 then
    a <= valor_expr_a_1;
  else
    a <= valor_expr_a_2;
  end if;
else
  if boolean_expr_3 then
    a <= valor_expr_a_3;
  else
    a <= valor_expr_a_4;
  end if;
end if;
```

1.b) (0.75 puntos) La sentencia **case** siguiente, suponiendo que `expr_case` tiene cinco posibles valores: `c0`, `c1`, `c2`, `c3` y `c4`.

```
case expr_case is
  when c0 =>
    a <= valor_expr_a_0;
    b <= valor_expr_b_0;
  when c1 =>
    a <= valor_expr_a_1;
    b <= valor_expr_b_1;
  when others =>
    a <= valor_expr_a_n;
    b <= valor_expr_b_n;
end case;
```

Pregunta 2 (3.5 puntos)

A continuación, se muestra la **entity** del circuito *restador completo de 1 bit*.

```
entity rest_completo is
  port ( res, acarreo_out : out std_logic;
        a, b, acarreo_in : in  std_logic );
end entity rest_completo;
```

- 2.a)** (1 punto) Escriba en VHDL una **architecture** que describa el *comportamiento* del circuito.
- 2.b)** (1 punto) Dibuje el diagrama del circuito restador completo de 1 bit al nivel de puertas lógicas. A continuación, escriba en VHDL la **entity** y la **architecture** de cada una de las puertas lógicas que componen el circuito que acaba de dibujar.
- 2.c)** (1.5 puntos) Escriba en VHDL una **architecture** que describa la *estructura* del circuito restador completo de 1 bit que ha dibujado, instanciando y conectando las puertas lógicas que ha diseñado anteriormente.

Pregunta 3 (3 puntos)

Se propone diseñar un circuito secuencial síncrono capaz de detectar cuándo recibe la secuencia “0100” por su entrada *x*. La **entity** del circuito se muestra a continuación.

```
entity detector is
  port( Y      : out std_logic;
        state : out std_logic_vector(2 downto 0);
        X      : in  std_logic;
        reset  : in  std_logic;
        clk    : in  std_logic );
end entity detector;
```

El circuito tiene una señal de reloj (*clk*), una entrada serie de un bit (*x*), una señal de reset asíncrona activa en ‘1’ (*reset*), una señal que indica el estado en que se encuentra el circuito (*state*) y una señal de salida de un bit (*Y*).

La señal *Y* se pone a ‘1’ si los últimos 4 bits recibidos por la entrada *x* se corresponden con la secuencia “0100”. La máquina no vuelve al estado inicial tras haber reconocido la secuencia, sino que detecta secuencias solapadas. La señal *reset* pone el circuito en su estado inicial. Todos los cambios tienen lugar en el flanco de subida de la señal de reloj.

Escriba en VHDL la **architecture** que describe el comportamiento del circuito en términos de una máquina de Moore. Dibuje el diagrama de estados correspondiente al circuito que ha diseñado.

Pregunta 4 (2 puntos)

Programe en VHDL un banco de pruebas para el circuito secuencial que ha diseñado al contestar a la Pregunta 3. El programa de test debe primero resetear el circuito y a continuación cargar en el circuito, a través de la entrada x, los siete bits siguientes (en el orden indicado): '0', '1', '0', '0', '1', '0', '0'. Si los valores de la señal de salida de la UUT no coinciden con lo esperado, el programa de test debe mostrar el correspondiente mensaje.