

INGENIERÍA DE COMPUTADORES III

INSTRUCCIONES

Por favor, entregue todas las hojas de enunciado junto con el examen.

Dispone de 2 horas para realizar el examen.

MATERIAL PERMITIDO: Ninguno.

Pregunta 1 (2 puntos)

Tomando como base el siguiente código VHDL, dibuje el cronograma de evolución de las señales x1, x2, x3, x4, x5 entre los instantes 0 y 60 ns.

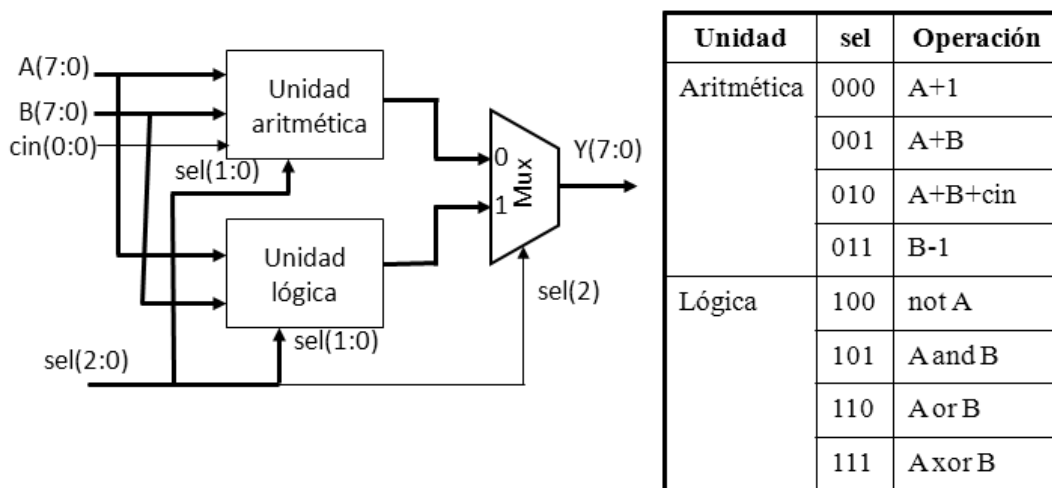
```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;
entity crono2 is
end entity crono2;

architecture crono2 of crono2 is
    signal x1 : unsigned (2 downto 0);
    signal x2, x3, x4, x5 : std_logic;
begin
    process is
    begin
        for i in 0 to 3 loop
            x1 <= to_unsigned(i,3);
            x2 <= std_logic(x1(2));
            x3 <= std_logic(x1(1));
            x4 <= std_logic(x1(0));
            wait for 10 ns;
        end loop;
        wait;
    end process;
    x5 <= x3 after 15 ns;
end architecture crono2;
```

Pregunta 2 (3 puntos)

A continuación, se muestra el circuito, la tabla de operaciones y la **entity** de una ALU. La ALU realiza operaciones sobre dos operandos de 8 bits, denominados A y B. La salida de la ALU se selecciona mediante el bit más significativo de la señal `sel`, mientras que la operación que realiza se especifica por los otros dos bits de esta señal. Escriba en VHDL la **architecture** que describe el comportamiento de la ALU, empleando para ello únicamente tres sentencias de asignación concurrente de selección. Asimismo, en el diseño únicamente pueden emplearse los dos siguientes paquetes de la librería IEEE:

```
IEEE.std_logic_1164
IEEE.numeric_std
```



```
entity ALU is
  port( Y      : out std_logic_vector ( 7 downto 0 );
        A, B   : in  std_logic_vector ( 7 downto 0 );
        sel    : in  std_logic_vector ( 2 downto 0 );
        cin    : in  std_logic_vector(0 downto 0));
end entity ALU;
```

Pregunta 3 (2.5 puntos)

Diseñe un circuito secuencial síncrono capaz de detectar cuando le llegan al menos tres ceros consecutivos por su entrada. La **entity** del circuito se muestra a continuación. El circuito tiene una señal de reloj (`clk`), un entrada serie de un bit (`X`), una señal de reset asíncrona activa en '0' (`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 por la entrada `X` se han recibido tres o más ceros consecutivos. 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.

```
entity detector is
  port( Y      : out std_logic;
        state : out std_logic_vector(1 downto 0);
        X      : in  std_logic;
        reset  : in  std_logic;
        clk    : in  std_logic);
end entity detector;
```

Pregunta 4 (2.5 puntos) Programe el banco de pruebas del circuito secuencial que ha diseñado. Explique detalladamente cómo el programa de test comprueba de manera sistemática el funcionamiento del circuito. El banco de pruebas debe comprobar que los valores obtenidos de la UUT coinciden con los esperados, mostrando el correspondiente mensaje en caso de que no coincidan. Al final del test, debe mostrarse un mensaje indicando el número total de errores.