

INGENIERÍA DE COMPUTADORES III

INSTRUCCIONES

Por favor, entregue esta primera hoja 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 y x6 entre los instantes 0 y 100 ns.

```
library IEEE;
use IEEE.std_logic_1164.all;

entity crono is
end entity crono;

architecture crono of crono is
    signal x1, x2, x3, x4, x5, x6 : std_logic;
begin
    x1 <= '0', '1' after 10 ns,
          '0' after 15 ns, '1' after 25 ns,
          '0' after 30 ns;
    x2 <= transport x1 after 8 ns;
    Procl: process(x1,x2)
    begin
        x3 <= x1 xor x2;
    end process;
    x4 <= x1 after 8 ns;
    x5 <= x1 xor x2;
    x6 <= x5;
end architecture crono;
```

Pregunta 2 (3 puntos)

Diseñe usando VHDL el circuito mostrado en la siguiente figura, que está compuesto por una unidad aritmético lógica (ALU) y un registro de desplazamiento. La ALU realiza operaciones sobre dos operandos de 8 bits, denominados A y B, y un operando de un bit denominado cin. La salida de la ALU es la entrada al registro de desplazamiento. El registro de desplazamiento opera en el flanco de subida de la señal de reloj CLK. La señal S, de 5 bits, determina la operación realizada por el circuito tal como se indica en las dos tablas de operaciones mostradas a continuación.

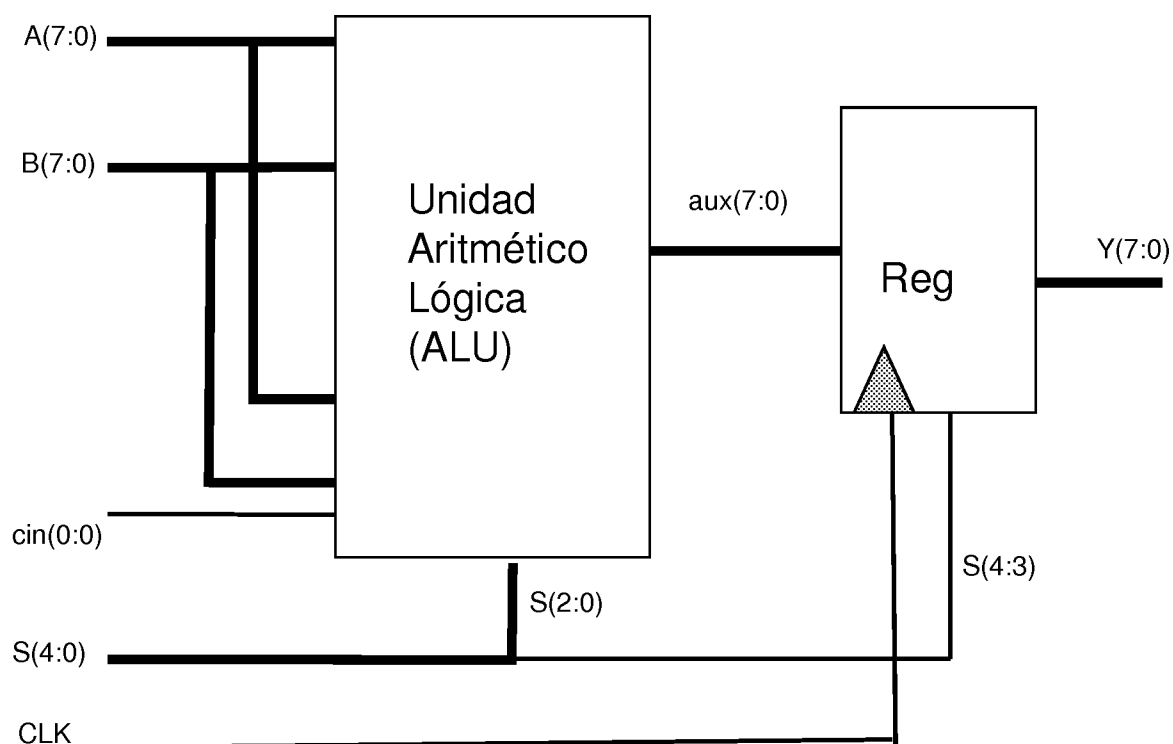


Tabla 1: Operaciones del registro.

S (4) S (3)	Operación
00	Carga de la señal aux desplazada un bit a la derecha, introduciendo un '0' en el bit más significativo
01	Carga de la señal aux desplazada un bit a la izquierda, introduciendo un '0' en el bit menos significativo
10	Carga de la señal aux rotada un bit a la derecha
11	Carga de la señal aux sin modificarla

Tabla 2: Operaciones de la ALU.

S (2) S (1) S (0)	Operación
000	A + B
001	A + B + cin
010	A + 1
011	B - 1
100	not B
101	A or B
110	A and B
111	A xor B

Escriba en VHDL la **architecture** que describe el comportamiento del circuito empleando un bloque **process** que describa el comportamiento de la ALU y otro bloque **process** que describa el comportamiento del registro. Asimismo, en el diseño únicamente pueden emplearse los dos siguientes paquetes de la librería IEEE:

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

El circuito ha de tener la **entity** siguiente:

```
entity ALUReg is
  port( Y      : out std_logic_vector ( 7 downto 0);
        A, B   : in  std_logic_vector (7 downto 0);
        cin    : in  std_logic_vector (0 downto 0);
        CLK    : in  std_logic;
        S      : in  std_logic_vector ( 4 downto 0));
end entity ALUReg;
```

Pregunta 3 (3 puntos)

Realice el diseño usando VHDL de un contador síncrono ascendente módulo 6. Es decir, la salida del circuito (señal `y`) toma cíclicamente los valores “000”, “001”, “010”, “011”, “100” y “101”. Si la cuenta está habilitada, se obtiene un nuevo valor de la salida en cada flanco de subida de la señal de reloj. El contador debe tener una entrada `c` activa a nivel alto que permita habilitar y deshabilitar la cuenta, la señal de reloj de entrada `clock` y una entrada `reset` síncrona activa a nivel alto, que pone la cuenta a '0'. Describa el comportamiento del circuito en términos de una máquina de Moore. Dibuje el diagrama de estados que describe el comportamiento del circuito.

Pregunta 4 (2 puntos)

Programe el banco de pruebas del circuito secuencial que ha diseñado en la Pregunta 3. 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.