

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.

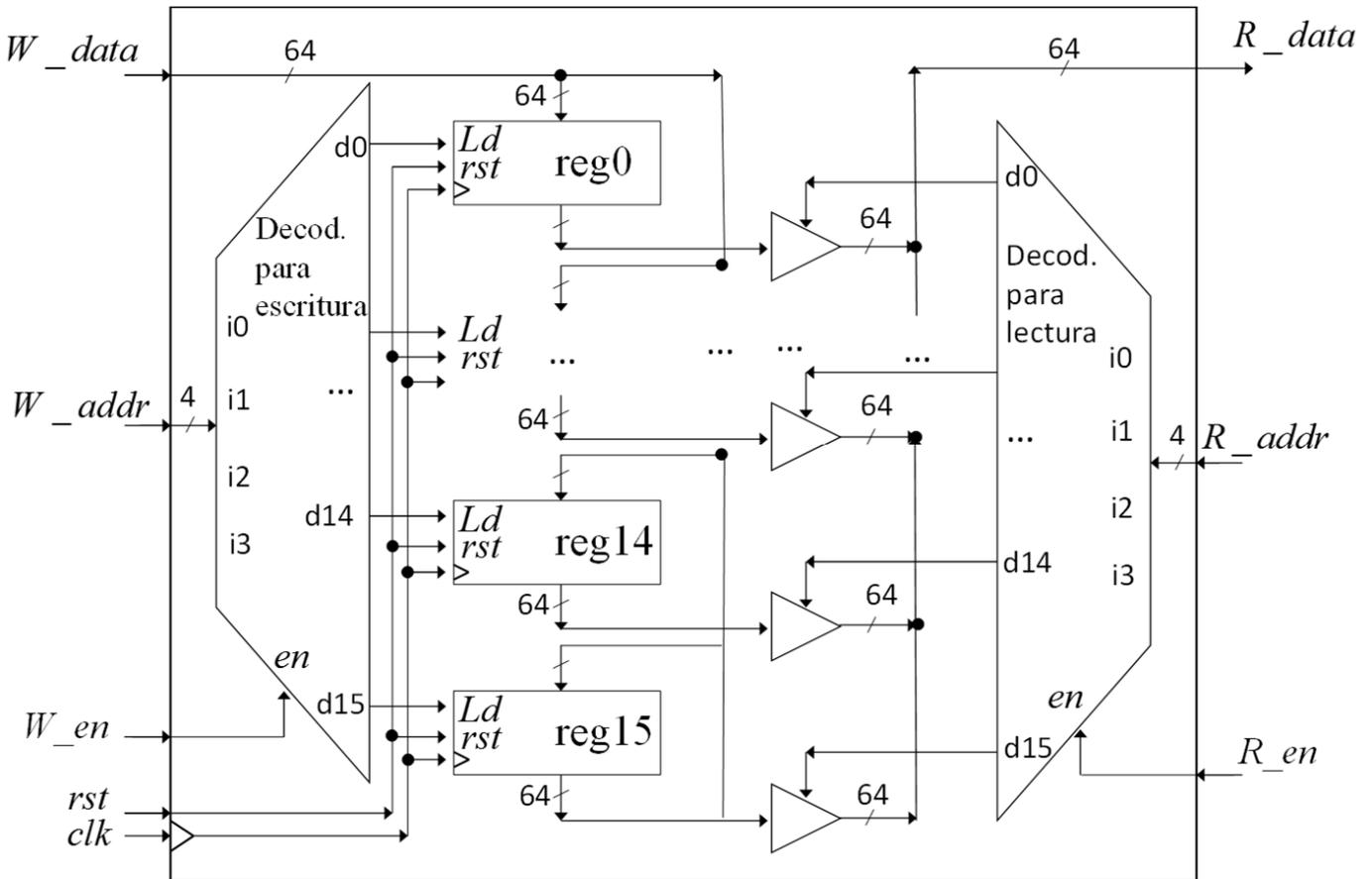
Pregunta 1 (2 puntos)

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 200 ns.

```
library IEEE;
use IEEE.std_logic_1164.all;
entity cronol is
end entity cronol;
architecture cronol of cronol is
    signal x1, x2 : std_logic := '0';
    signal x3, x4, x5, x6 : std_logic;
begin
    process (x1)
        variable temp1, temp2, temp3: std_logic;
    begin
        if (rising_edge(x1)) then
            temp1 := x2;
            temp2 := temp1;
            temp3 := temp2;
            x3 <= temp3;
            x4 <= x2;
            x5 <= x4;
            x6 <= x5;
        end if;
    end process;
    x1 <= not x1 after 20 ns;
    x2 <= '0', '1' after 30 ns, '0' after 90 ns;
end architecture cronol;
```

Pregunta 2 (3 puntos)

Se pretende diseñar un *register file* de tamaño 16*64 que opera en el flanco de subida de la señal de reloj. Es decir, que contiene 16 registros, cada uno de los cuales tiene 64 bits. Este circuito se puede diseñar empleando dos decodificadores, 16 registros de 64 bits y 16 buffer triestado de 64 bits, tal como se muestra en la siguiente figura.



El circuito tiene una señal de salida R_data de 64 bits y una señal de entrada W_data de 64 bits.

Tiene asimismo una señal síncrona de reset a nivel alto llamada rst y una señal de reloj clk . La señal de reset pone todos los bits de todos los registros a cero.

Las señales de entrada W_en y R_en se emplean para habilitar la escritura y la lectura, respectivamente.

Las señales de entrada W_addr y R_addr se emplean para direccionar uno de los 16 registros.

Cuando la señal rst tiene valor '0' y la señal W_en tiene valor '1', el registro cuya dirección sea W_addr se carga en el flanco de subida de la señal de reloj con la señal de entrada W_data .

Cuando la señal `R_en` tiene valor '1', la señal de salida `R_data` toma el valor de la salida del registro cuya dirección sea `R_addr`. Por el contrario, cuando la señal `R_en` tiene valor '0', la señal de salida `R_data` toma el valor de alta impedancia.

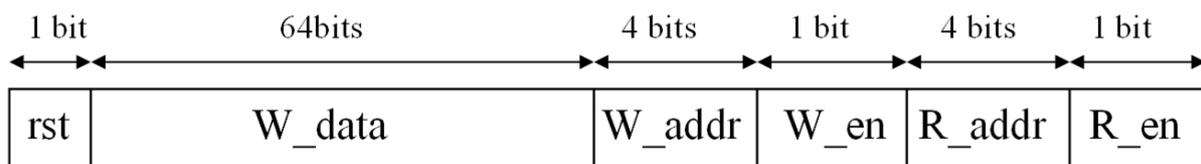
La **entity** del circuito se muestra a continuación.

```
entity RegFile16x64 is
  port ( R_data : out std_logic_vector(63 downto 0);
        W_data : in  std_logic_vector(63 downto 0);
        R_addr, W_addr: in std_logic_vector(3 downto 0);
        R_en, W_en: in std_logic;
        clk, rst: in std_logic );
end entity RegFile16x64;
```

Escriba en VHDL la **architecture** que describe el comportamiento del *register file* empleando para ello dos bloques **process**.

Pregunta 3 (2.5 puntos)

Diseñe un banco de pruebas para el *register file* que ha diseñado en la Pregunta 2 que permita realizar una inspección visual de las señales de entrada y de salida del circuito. El banco de pruebas debe generar una señal de reloj de periodo 10 ns y valor inicial '0'. El resto de valores de las señales de entrada los ha de leer de un fichero llamado *vectores.txt*. Cada una de las líneas de este fichero de texto consta de una palabra de 75 bits cuyo significado se muestra en la siguiente figura.



Pregunta 4 (2.5 puntos)

Diseñe un circuito secuencial síncrono capaz de detectar si el número de unos que le ha llegado por su entrada serie de un bit es par o impar. La **entity** del circuito se muestra a continuación.

```
entity detector is
  port( Y      : out std_logic;
        X      : in  std_logic;
        reset  : in  std_logic;
        clk    : in  std_logic);
end entity detector;
```

El circuito tiene una señal de reloj (`clk`), un entrada serie de un bit (`X`), una señal de reset asíncrona activa en '1' (`reset`) y una señal de salida de un bit (`Y`).

La señal `reset` pone el circuito en su estado inicial. Este estado inicial indica que no ha llegado ningún '1' por su entrada serie. Es decir, que el número de unos que ha llegado hasta el momento es un número par. Todos los cambios tienen lugar en el flanco de subida de la señal de reloj.

La señal `Y` se pone a '1' sólo si por la entrada `X` ha llegado un número impar de unos. Por el contrario, si el número de unos que ha llegado por la entrada `X` es par, la señal `Y` se pone a '0'.

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.