2-4-2024

ANDRADE SALAZAR, IGNACIO

Centro Universitario de los Valles, Universidad de Guadalajara

Programación de sistemas reconfigurables

Tarea 7. Registros y contadores

Ingeniería en electrónica y computación

**Registros**

**1 En VHDL, el concepto.... complete todo el párrafo**

En VHDL, el concepto de registro hace referencia a la transferencia de información que se da en la interconexión de varios flip-flops acomodados en un arreglo en serie o en paralelo, o inclusive ambos.

**2 Registro serie, explique el funcionamiento e incluya la figura 3.12a y 3.12b**

Considérese el arreglo que se muestra en la figura 3.12 a). Como se puede observar en este, la actualización de la salida en cada uno de los flip-flop se da de manera simultanea en cada flanco de subida de la señal de reloj (clk); la entrada **d** se transfiere a la salida **a,** esta se transfiere a **b;** por último, **b** se transfiere a **q.** Como podemos ver, los pins de entrada y de salida son d y q, respectivamente. En este caso, la transferencia de información se realiza mediante las señales a y b, “signal a y b”, mientras que la asignación de información se produce de la línea 13 a la línea 15. Nótese, en la declaración de los puertos, que solo se ha asignado pin a la entrada **d** y a la salida **q.**

library ieee;

use ieee.std\_logic\_1164.all;

entity registro is

port (d, clk: in std\_logic;

q: out std\_logic);

end registro;

architecture serie of registro is

signal a,b: in std\_logic;

begin

process (clk)

begin

if ( clk'event and clk='1' ) then

a<=d;

b<=a;

q<=d;

end if;

end serie;

Imagen que contiene texto, pizarrón

Descripción generada automáticamente

**3 Variables explique ampliamente el concepto**

Una variable solo puede ser declarada en las sentencias secuenciales process o subprogramas, Una variable básicamente puede considerarse como un elemento de memoria, “dato”, al que se asigna un valor inicial o el valor final incremental dentro de la síntesis de un programa; su formato es el siguiente:

Variable <nombre>: <tipo>:= <expresión>

Variable a: integer:= 16;

**4 Registro paralelo, explique el funcionamiento e incluya la figura 3.14a, 3.14b y 3.14c**

La función de un registro de entrada-salida en paralelo es la transferencia de la información de sus pins de entrada a sus pins de salida, en la transición de un flanco de subida o uno de bajada en un pulso de reloj. En la figura 3.14a) se ilustra la conexión de 8 flip-flop tip D; como se observa, la señal de reloj se aplica de forma simultanea a cada uno de estos. Por su parte en la figura 3.14b) se utiliza un vector para describir su comportamiento. En tanto, el código correspondiente de la figura 3.14c9 muestra que la entrada D transfiere la información a la salida Q en el flanco de subida de la señal de reloj clk.

library ieee;

use ieee.std\_logic\_1164.all;

entity reg is port (

D: in std\_logic\_vector (0 to 7);

clk: in std\_logic;

Q: out std\_logic\_vector 80 to 7));

end reg;

architecture arqreg of reg is

begin

process (clk) begin

if (clk'event and clk='1' ) then

Q <= D;

end if;

end process;

end arqreg;

Diagrama, Esquemático

Descripción generada automáticamente

Imagen que contiene Diagrama

Descripción generada automáticamente

**3.3. Contadores**

**1. Favor de transcribir el párrafo que inicia con: Los contadores son....**

Los contadores son entidades muy utilizadas en el diseño lógico, que hacen uso extensivo de los operadores lógicos (+ y -) para realizar operaciones de incremento o decremento de datos.

**2. Contador con reset y enable: Ponga la figura 3.20 el dispositivo y el código. Además, mencione para que se necesita la librería: use work.std\_arith.all**

library ieee;

use ieee.std\_logic\_1164.all;

use work.std\_arith.all;

entity contador is

port ( clk, reset: in std\_logic;

q: inout std\_logic\_vector ( 3 downto 0));

end contador;

architecture modulo of contador is

begin

process (clk, reset)

begin

if ( clk'event and clk='1' ) then

if (reset = '1' or q="1001" ) then

q<=q+1;

end if;

end if;

end process;

end modulo;

La librería work.std\_arith se utiliza en la arquitectura cuando se le suma 1 a q ya que la suma forma parte de esta librería en VHDL.

Imagen que contiene Gráfico

Descripción generada automáticamente

**3. Poner la imagen de la Figura 3.21 y el código, además explique a línea 6**

library ieee;

use ieee.std\_logic\_1164.all;

use work.std\_arith.all;

entity cont is

port ( clk, reset: in std\_logic;

Q: inout integer range 0 to 15;

end cont;

architecture arq\_cont of cont is

begin

process (clk, reset) begin

if (clk'event and clk='1' ) then

if (reset='1' or Q=9 ) then

Q<=0;

else

Q <= Q+1;

end if;

end if;

end process;

end arq cont;

En la línea 6 se establece la variable Q como entrada y salida, así como con un valor de entero entre 0 y 15, es decir que producirá el conteo correspondiente a los valores 0, 1, 2 hasta el 15.

Imagen que contiene texto, pizarrón

Descripción generada automáticamente

**4. Explique cómo funciona un contador ascendente-descendente y ponga el listado 3.7**

La señal de control up/down permite definir si el conteo se realiza de manera ascendente o descendente. En este caso, cero aplicado a esta señal determina una cuenta ascendente, es decir, de 0 a 15. De esta forma, el funcionamiento del circuito queda determinado por dos señales: el pulso de reloj (clk) y la señal up/down, como se muestra en el siguiente programa, donde se puede observar que en este caso se hace uso del operador + y -, respectivamente.

library ieee;

use ieee.std\_logic\_1164.all;

use work.std\_arith.all;

entity contador is

port ( clk: in std\_logic;

UP: in std\_logic;

Q: inout std\_logic\_vector(3 downto 0));

end contador;

architecture a\_contador of contador is

begin

process (UP = '0') then

if (clk'event and clk='1') then

if (UP = '0') then

Q <= Q+1;

else

Q <= Q-1;

end if;

end if;

end process;

end a\_contador;