

Introducción al VHDL: Plancha de problemas

1)

a) Analizar el código propuesto y el circuito que generaría el ambiente de diseño con la herramienta "View Schematic RTL"

b) Reescribir el código definiendo D como una variable y comparar los circuitos que el ambiente de diseño generaría para ambos casos.

c) Analizar los circuitos resultantes y explicar las diferencias que se observan.

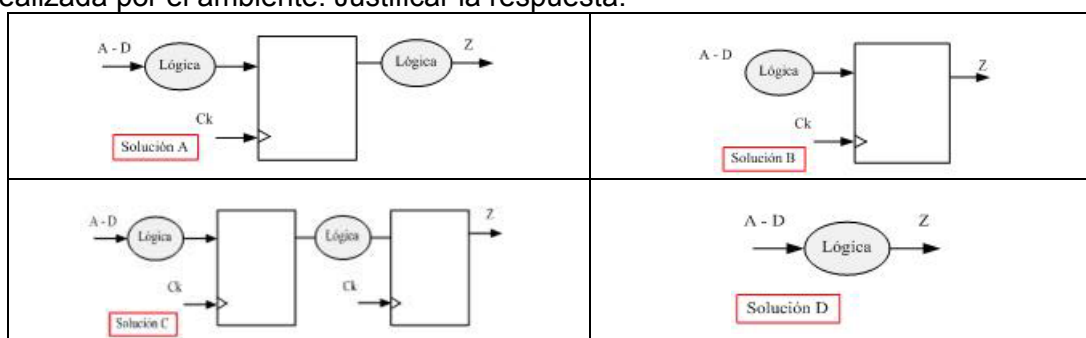
```
Library IEEE;
use IEEE.std_logic_1164.all;
entity xor_sig is
port (A, B, C: in STD_LOGIC;
      X, Y: out STD_LOGIC);
end xor_sig;
architecture SIG_ARCH of xor_sig is
signal D: STD_LOGIC;
begin
process (A,B,C)
begin
D <= A;
X <= C xor D;
D <= B;
Y <= C xor D;
end process;
end SIG_ARCH;
```

2) Proponer una entidad con puertos A, B, C, D (entrada) y Z (salida).

a) En la arquitectura asociada, definir un proceso sensible al Ck en el cual conste:

```
If ck'event and ck = '1' Then
X <= A + B;
Y <= C + D;
Z <= X + Y;
End if;
```

b) Analizar los cuatro esquemas propuestos y determinar cual sería el resultante de la síntesis realizada por el ambiente. Justificar la respuesta.



c) Introducir el código en el ambiente y realizar la síntesis para verificar la respuesta anterior.

3) Dadas las seis arquitecturas que se describen a continuación:

1) Architecture rtl1 of exer is Signal i1: std_logic; Begin Process (clk) begin i1 <= d and a; q2 <= i1 and b; end process ; End ;	2) Architecture rtl2 of exer is Signal i1, i2: std_logic; Begin Process (clk) begin i1 <= d and a; i2 <= i1; end process ; q2 <= i2 and b; End ;	3) Architecture rtl3 of exer is Signal q1, i1, i2: std_logic; Begin Process (clk) begin q1 <= d; i1 <= q1 and a; i2 <= i1; q2 <= i2 and b; end process ; End ;
4) Architecture rtl4 of exer is Signal q1, q2: std_logic; Begin Process (clk) begin q1 <= d; q2 <= q1 and a; end process ; i2 <= q2 and b; End ;	5) Architecture rtl5 of exer is Signal q1, i1, i2: std_logic; Begin Process (clk) begin q1 <= d; q2 <= i1; end process ; i2 <= q2 and b; i1 <= q1 and a; End ;	6) Architecture rtl6 of exer is Signal q1, q2: std_logic; Begin Process (clk) begin if clk'event and clk='1' then q1 <= d; q2 <= q1 and a; end if ; end process ; i2 <= q2 and b; End ;

- a) Dibujar para cada caso la entidad asociada (con sus puertos).
 - b) Indicar el circuito que implementará el ambiente. Justificar.
 - c) Verificar sus conclusiones volcando el código correspondiente.
- 4) Describir en VHDL un contador de 8 bits, con reset asincrónico, señal de enable de Ck, que sea capaz de cargar un valor inicial en paralelo y de contar en forma ascendente y descendente. Asignar a las señales el siguiente orden de prioridad: reset, load, enable / Up-Down.
- a) Modificar el diseño para que el reset sea sincrónico y para que al contar incremente y decremente de a dos.
 - b) Modificar ambos diseños para definir sendos contadores de n bits, donde n se pueda especificar de acuerdo a la necesidad de utilización.
 - c) Verificar el funcionamiento utilizando la herramienta de simulación disponible.
- 5) Describir en estilo estructural un registro de desplazamiento a la izquierda de 4 bits que responda al mostrado en la figura utilizando FF tipo D de la biblioteca Unisim. Verificar el funcionamiento mediante simulación.

