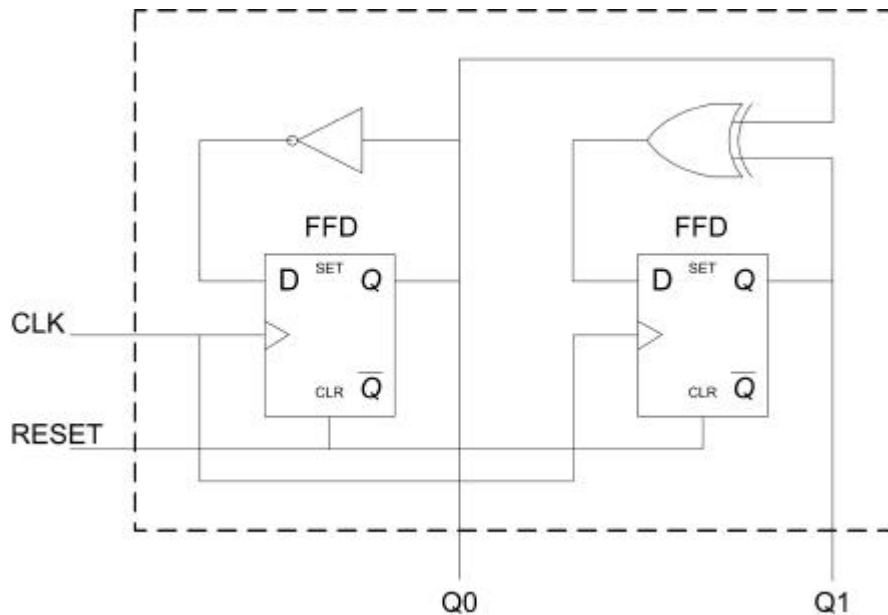


Sistemas Digitales (66.17)

Práctica 1 - VHDL

1. Implementar en VHDL un contador de 2 bits de manera estructural (indicar en el dibujo todas las señales utilizadas). Simular.



2. Realizar el diagrama en bloques de un contador de 4 bits e implementarlo en VHDL (indicar en el dibujo todas las señales utilizadas). Implementar la entidad de simulación. Simular.
3. Describir en VHDL, por el modelo de comportamiento, la función lógica dada por el siguiente Karnaugh. Simular.

		X1 X2			
		00	01	11	10
X3 X4	00	0	1	1	0
	01	1	1	0	0
	11	0	0	0	1
	10	1	0	0	1

4. Confeccionar el Karnaugh representativo del siguiente código. Luego describirlo utilizando la sentencia if-else.

```

entity karnaugh is
  port (x: in bit_vector(0 to 3); z: out bit);
end;

architecture beh of karnaugh is
  signal w1, w2, w3, w4: bit;
begin
  process(x)
  begin
    case x is
      when "0100" | "0101" | "1100" | "1010" | "1011" | "0010"
        => z <= '1';
      when others => z <= '0';
    end case;
  end process;
end;

```

5. La siguiente tabla de flujo representa la descripción de un circuito secuencial asincrónico. Realizar la descripción en VHDL de dicho circuito. Simular.

	00	01	11	10
a	a,0	b	a,0	a,0
b	a	b,0	c	-
c	d	c,1	c,1	c,1
d	d,1	c	-	a

6. La siguiente tabla de flujo representa la descripción de un circuito secuencial asincrónico. Realizar la descripción en VHDL de dicho circuito. Simular.

	00	01	11	10
a	a,0	b,1	a,0	a,1
b	a,1	b,0	b,1	b,0

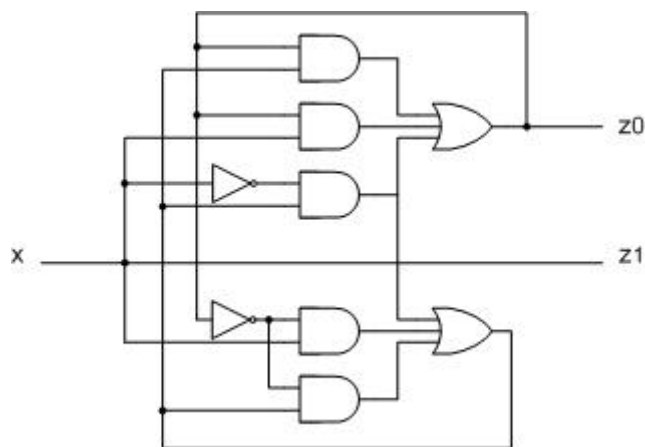
7. Describir en VHDL un circuito síncronico de una entrada y una salida, tal que por cada pulso de reloj la salida valga 1 si los últimos 4 bits fueron “1001”. En la entrada nunca pueden existir más de dos 1 consecutivos. Simular.
8. Describir en VHDL un circuito asincrónico de dos entradas y dos salidas. En estas últimas se deberá indicar cuál fue la última entrada en cambiar. Sólo puede cambiar una entrada por vez, y las únicas salidas posibles son “01” y “10”. Simular.

	00	01	11	10
a	a,10	c	-	b
b	a	-	d	b,10
c	f	c, 01	e	-
d	-	g	d, 01	h
e	-	g	e, 10	h
f	f, 01	c	-	b
g	f	g,10	e	-
h	a	-	d	h,01

9. Describir en VHDL un contador de flancos (ascendentes y descendentes) módulo 4.

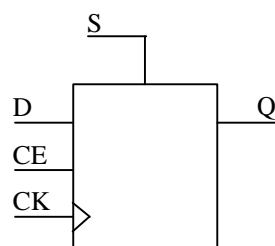
	0	1
a	a,00	b,--
b	c,--	b,01
c	c,10	d,--
d	a,--	d,11

Verificar que el siguiente circuito cumple con lo pedido (implementándolo en VHDL):



10. Describir en VHDL un Flip Flop D de flanco ascendente con set y reset asincrónicos.

11. Implementar en VHDL un registro de 4 bits con flanco positivo de clock, con set asincrónico y clock enable.



12. Implementar en VHDL (indicar en el dibujo todas las señales utilizadas) los distintos circuitos integrados (Dual JK Flip Flop, Dual 4 Input Multiplexer, SIPO Shift Register, 9 Bit Parity Checker, 4 Bit Full Adder With Fast Carry), cuyas hojas de datos se encuentran en la página de la materia (<http://cactus.fi.uba.ar/6617>). Implementar las correspondientes entidades de simulación. Simular.

13. Explicar los siguientes códigos:

```
for J in 10 downto 5 loop
    if SUM < TOTAL_SUM then
        SUM := SUM + 2;
    elsif SUM = TOTAL_SUM then
        next;
    else
        null;
    end if;
    K := K + 1;
end loop;
```

CODIGO 1

```
L1: for J in 0 to 10 loop
    ... -- grupo de sentencias 1
    L2: loop
        ... -- grupo de sentencias 2
        next L1 when isDone = '1';
        ... -- grupo de sentencias 3
    end loop L2;
    ... -- grupo de sentencias 4
end loop L1;
```

CODIGO 2

Qué sucedería si en el código 2 se reemplazara la línea:

“next L1 when isDone = '1';”

por la sentencia:

“exit L2 when isDone = '1';”