

EL253 - Sistemas Digitales

Semestre 2022-1

Profesor Kalun José Lau Gan

Sesión de Teoría Semana 4

1

Preguntas previas: (mantengan la mano alzada durante la intervención)

- He conseguido una fuente de 5V/2A en lugar de una de 1A, funcionará para la implementación?
 - Si, una fuente de mayor amperaje no influye en el funcionamiento de los circuitos que estamos implementando, tendremos mayor “holgura”.
- Veremos mas implementaciones en el curso?
 - Todas las sesiones vemos implementación
- Qué va a venir en la PC1?
 - Todo lo visto hasta la sesión anterior a la aplicación de dicha evaluación
- Será grupal la PC1?
 - No!

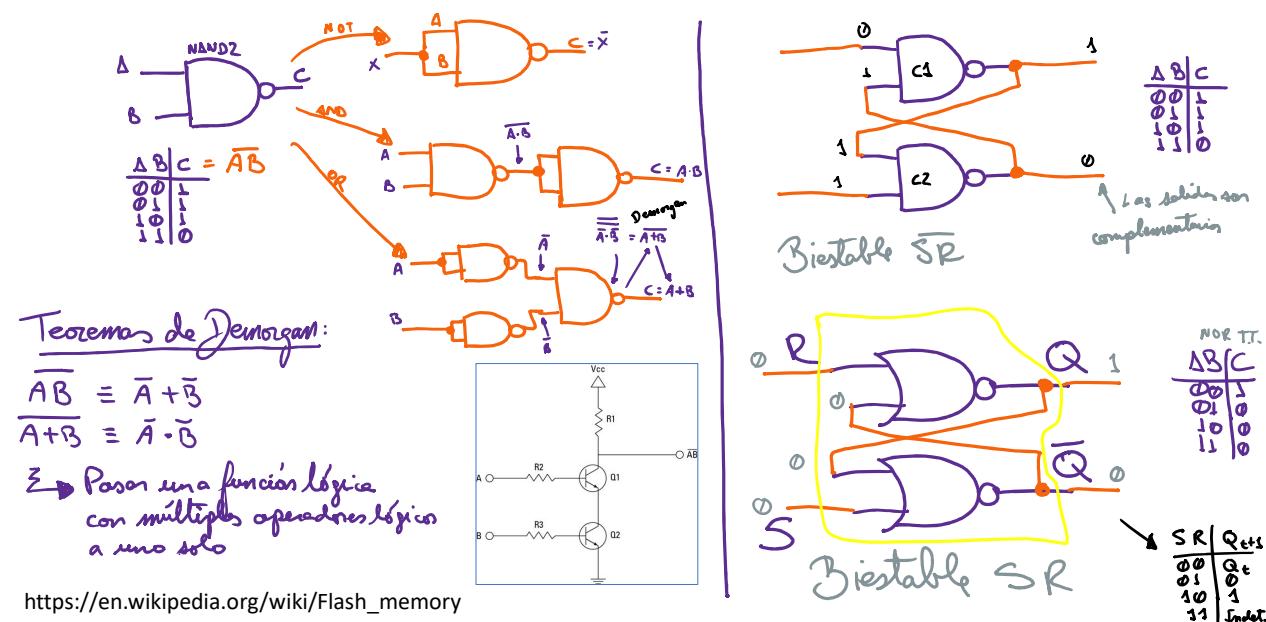
2

Agenda

- Sistemas secuenciales
- Flanco vs Nivel
- Modelo de contador binario en VHDL
- Módulo divisor de frecuencia en VHDL
- Descripción estructural en VHDL

3

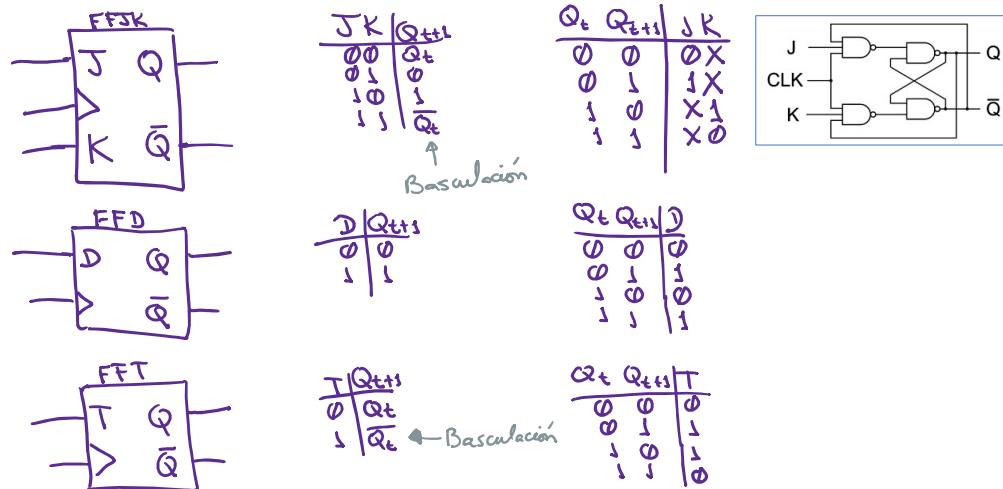
Introducción a los sistemas secuenciales



4

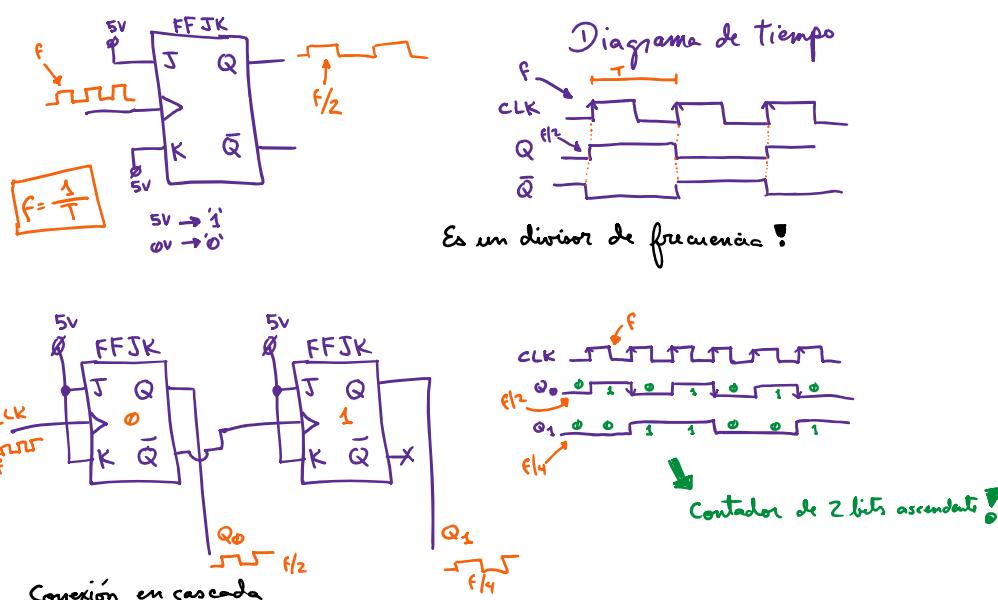
Sistemas Secuenciales

- Un nuevo escenario: el tiempo
- Flip-flops como unidad mínima de un sistema secuencial (SR, JK, T, D)



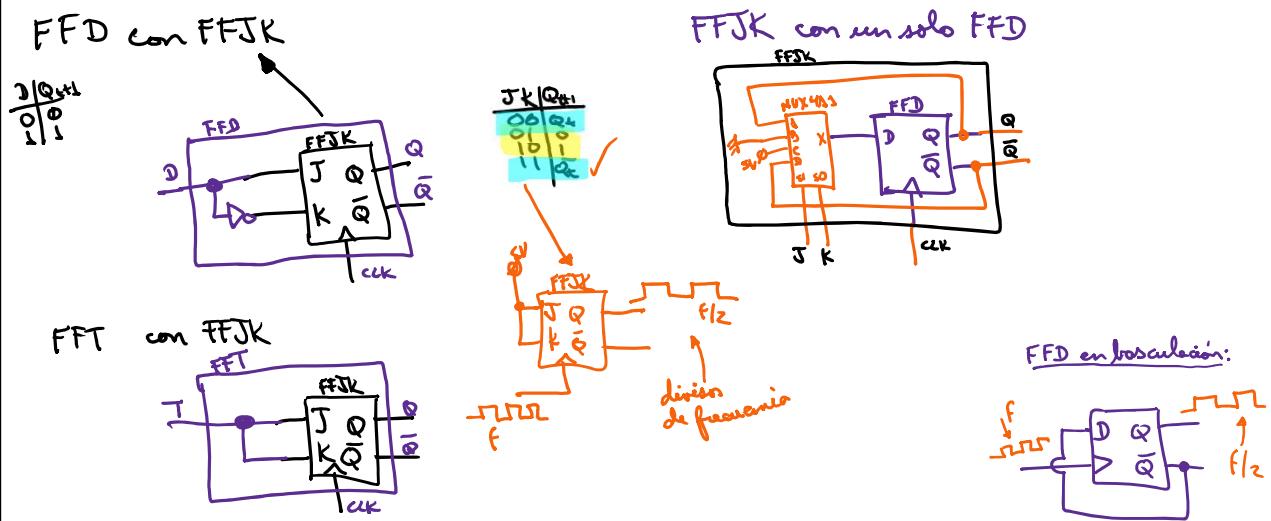
5

Concepto de “basculación”



6

Conversión entre tipos de FFs:



7

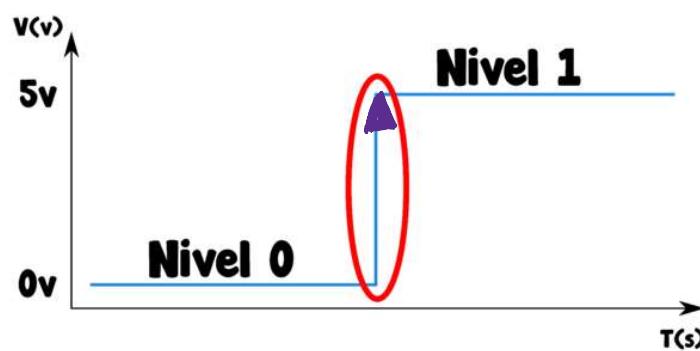
Receso:



8

Flanco vs nivel lógico

- Flanco es la transición de un nivel a otro nivel lógico
 - Transición positiva (paso de nivel 0 a nivel 1)
 - Transición negativa (paso de nivel 1 a nivel 0)



9

Cuenta personas

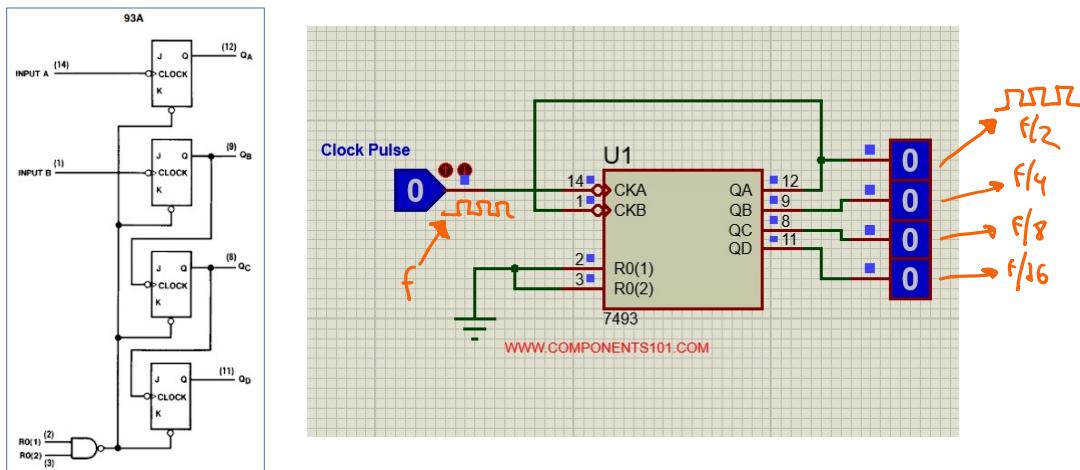
- Sirve para controlar el aforo



10

Contadores binarios

- Circuitos secuenciales
- Tienen un registro interno donde se almacena la cuenta



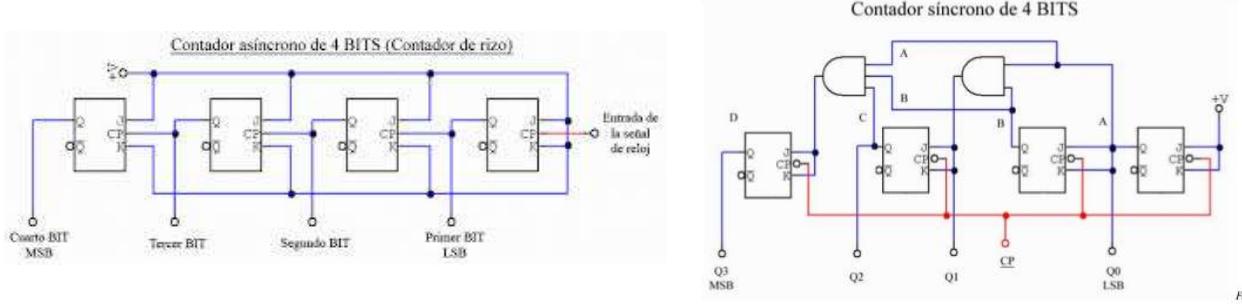
11

Sobre contadores:

- Los contadores pueden tener uno o mas entradas de reloj
- Las entradas de reloj pueden ser **positivas o negativas**
- Poseen un registro de cuenta
- Hay contadores de cuenta ascendente y de cuenta descendente
- Cuando la cuenta se encuentre en el valor máximo (ó mínimo) se producirá un desborde (overflow) y se regresara al valor inicial (ó final)
- Pueden tener entradas adicionales de control (Enable, Reset, Load, etc)
- Pueden tener un puerto de entrada de datos para precargar la cuenta.
- Pueden contar de uno en uno o de manera personalizada.
- En VHDL no se especifica si va a ser asíncrono o síncrono.

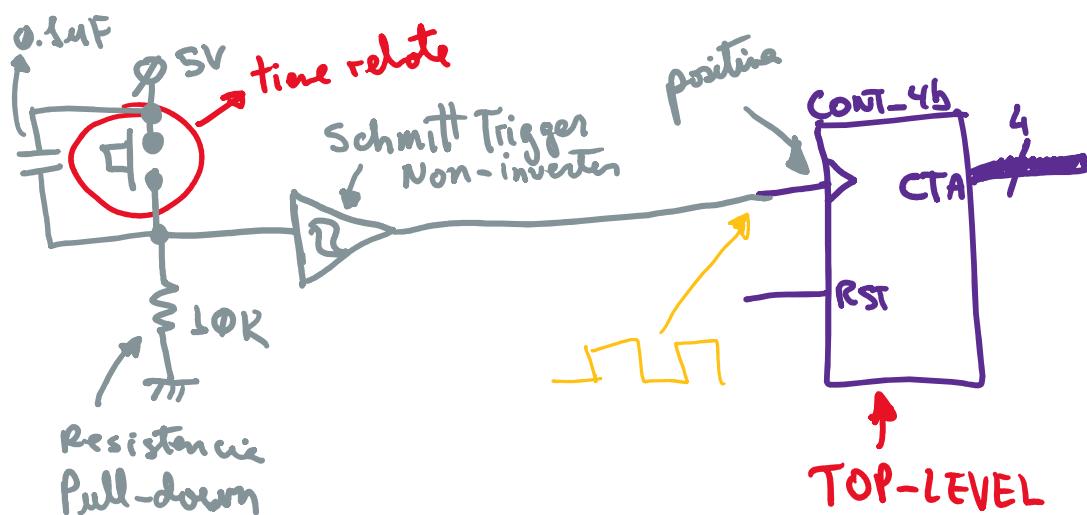
12

Contador síncrono vs asíncrono:



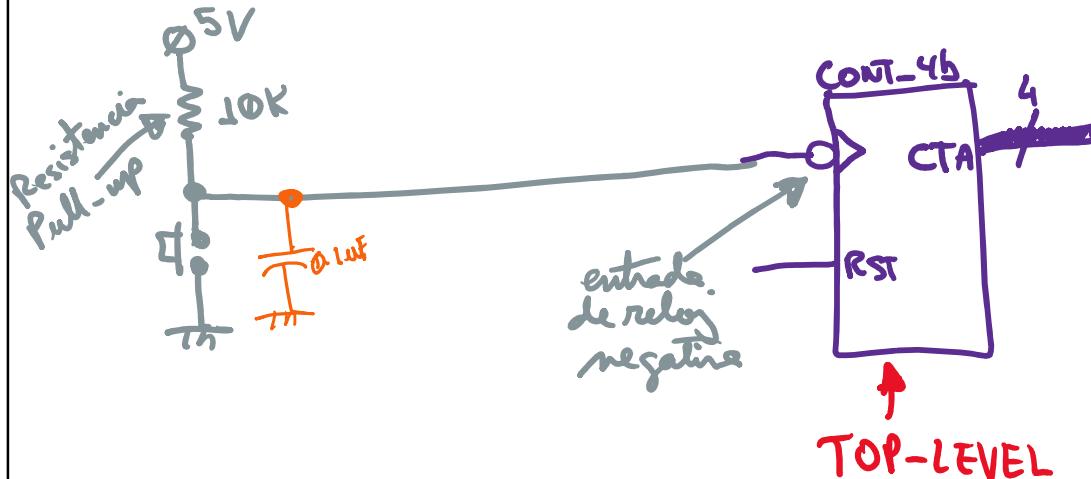
13

Modelo de contador binario básico de 4 bits ascendente con entrada de reset y reloj positivo



14

Modelo de contador binario básico de 4 bits ascendente con entrada de reset y reloj negativo



15

Código en VHDL del contador básico binario:

```

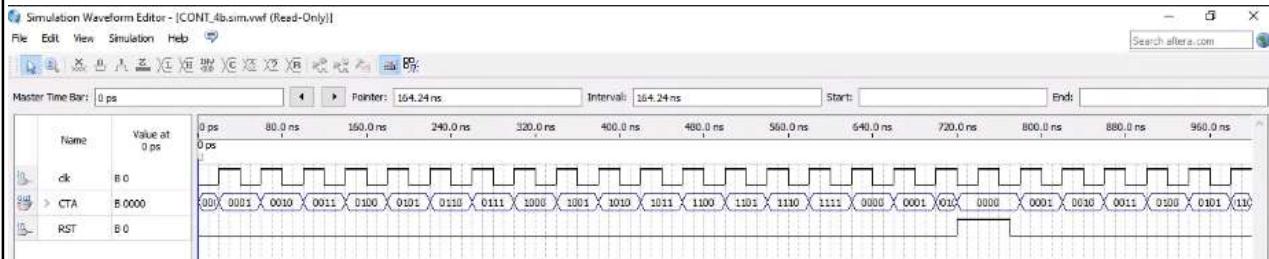
entity CONT_4b is
  port(clk, RST: in std_logic;
       CTA: out std_logic_vector(3 downto 0));
end CONT_4b;

architecture Oyuco of CONT_4b is
signal cta_interno: std_logic_vector(3 downto 0);
begin
  CTA <= cta_interno;
  process(clk, RST)
  begin
    if RST = '1' then
      cta_interno <= (others => '0');
    else
      if rising_edge(clk) then
        if cta_interno = "1111" then
          cta_interno <= (others => '0');
        else
          cta_interno <= cta_interno + 1;
        end if;
      end if;
    end process;
  end Oyuco;

```

16

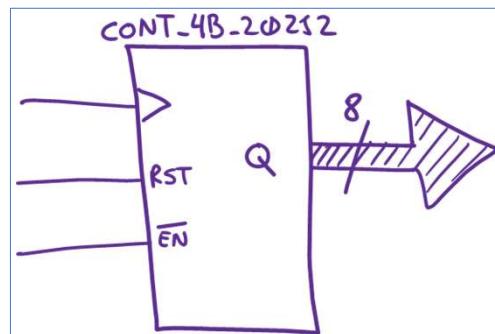
Simulación en VWF:



17

Ejercicio: Contador de 8 bits de cuenta ascendente con entrada positiva de reloj, reset activo en alto y habilitador de salidas activo en bajo

- El habilitador colocará las salidas en alta impedancia cuando esté en uno lógico pero la cuenta seguirá funcionando.
- El reset es el mecanismo que reiniciará la cuenta a cero si es está en uno lógico.



18

Descripción en VHDL:

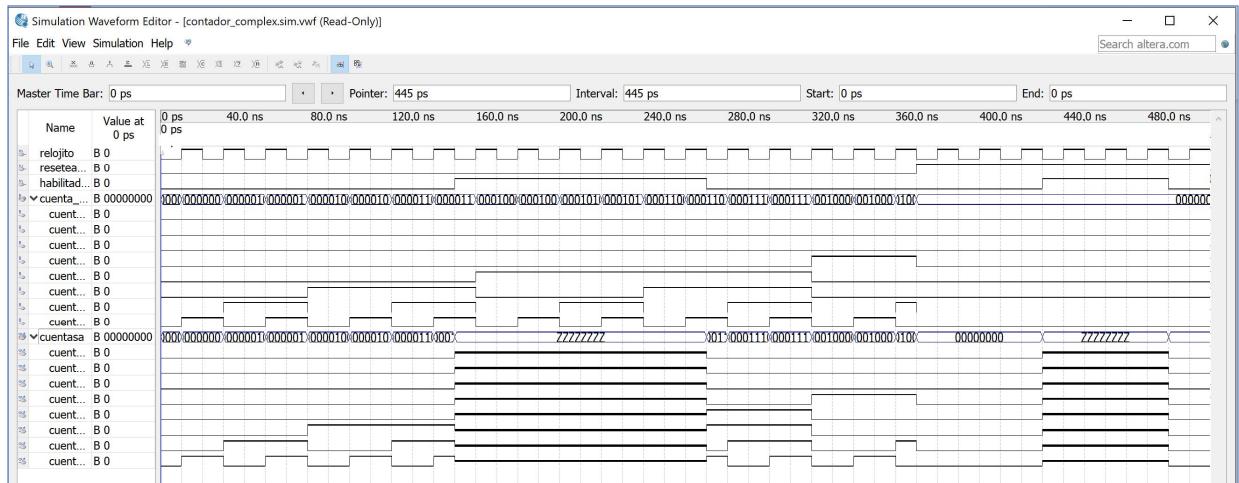
```

1 library IEEE;
2 use IEEE.std_logic_1164.all;
3 use IEEE.std_logic_arith.all;
4 use IEEE.std_logic_unsigned.all;
5
6 entity contador_complex is
7   port( relojito,reseteador, habilitador:  in std_logic;
8         cuenta:    out std_logic_vector(7 downto 0));
9 end contador_complex;
10
11 architecture funcionamiento of contador_complex is
12 signal cuenta_interna:  std_logic_vector(7 downto 0);
13 begin
14   process(relojito,reseteador)
15   begin
16     if reseteador = '1' then
17       cuenta_interna <= (others => '0');
18     else
19       if rising_edge(relojito) then
20         cuenta_interna <= cuenta_interna + 1;
21       end if;
22     end if;
23   end process;
24   with habilitador select
25     cuenta <= cuenta_interna when '0',
26     (others => 'Z') when others;
27 end funcionamiento;

```

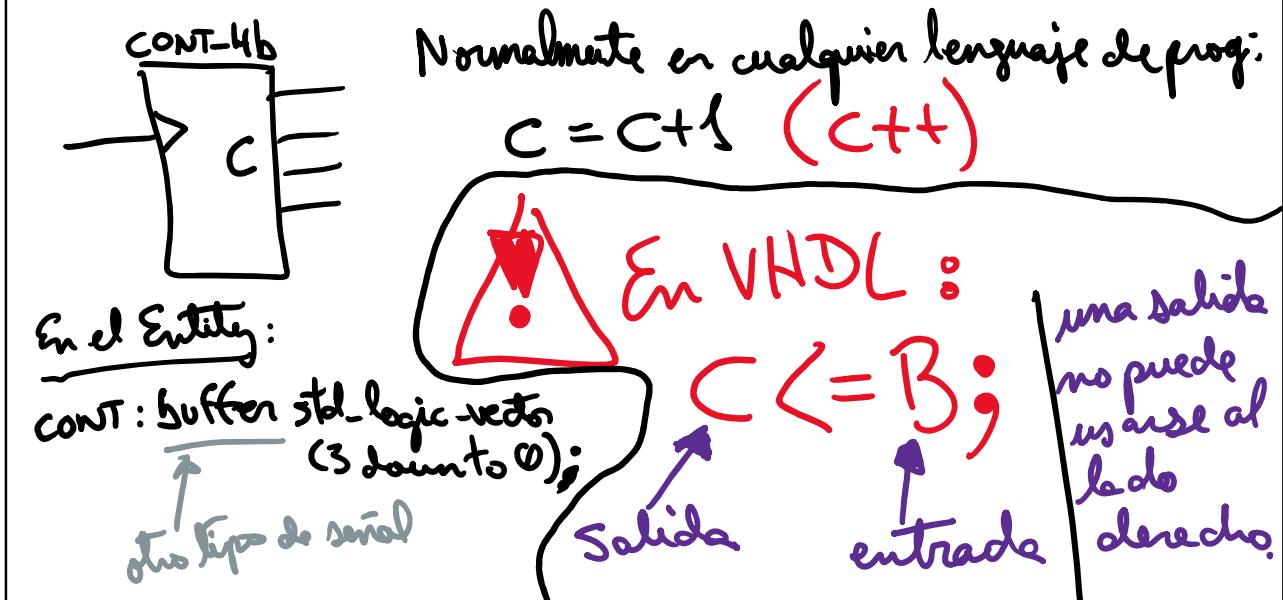
19

Simulación en diagramas de formas de onda:



20

¿Por qué se está usando una señal interna?

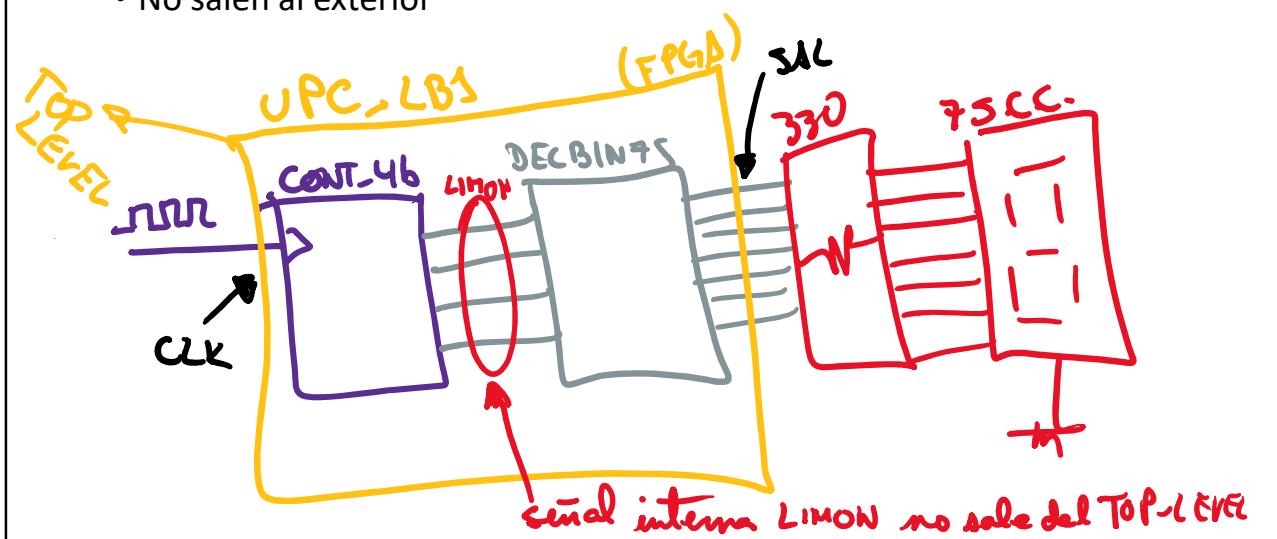


21

Señales internas en VHDL

- Se declaran dentro de la arquitectura
- No salen al exterior

Entity { C2K IN
UPC_LBS SSL OUT }



22

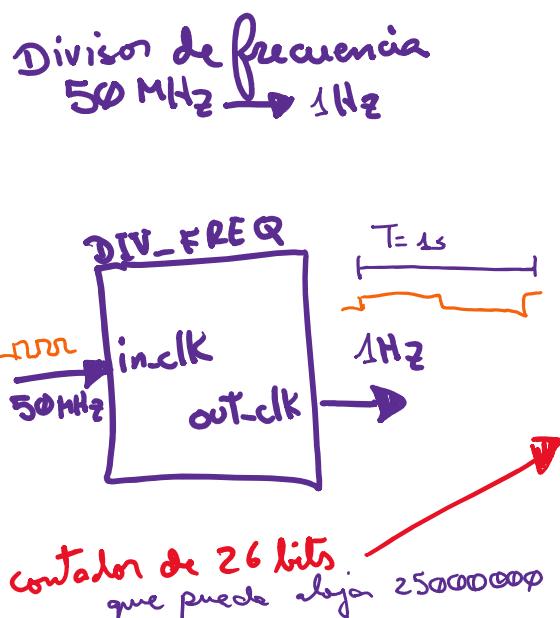
Divisor de frecuencia

- Se emplea para disminuir la frecuencia de una señal de reloj
- En nuestro caso tenemos que usar un divisor de frecuencia para disminuir el reloj de 50MHz que hay a la entrada del FPGA de la tarjeta EP2C5T144
- Usualmente es un contador binario ($f/2$, $f/4$, $f/8$, etc)



23

Modelo del divisor de frecuencia en VHDL



```

use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
use IEEE.std_logic_unsigned.all;

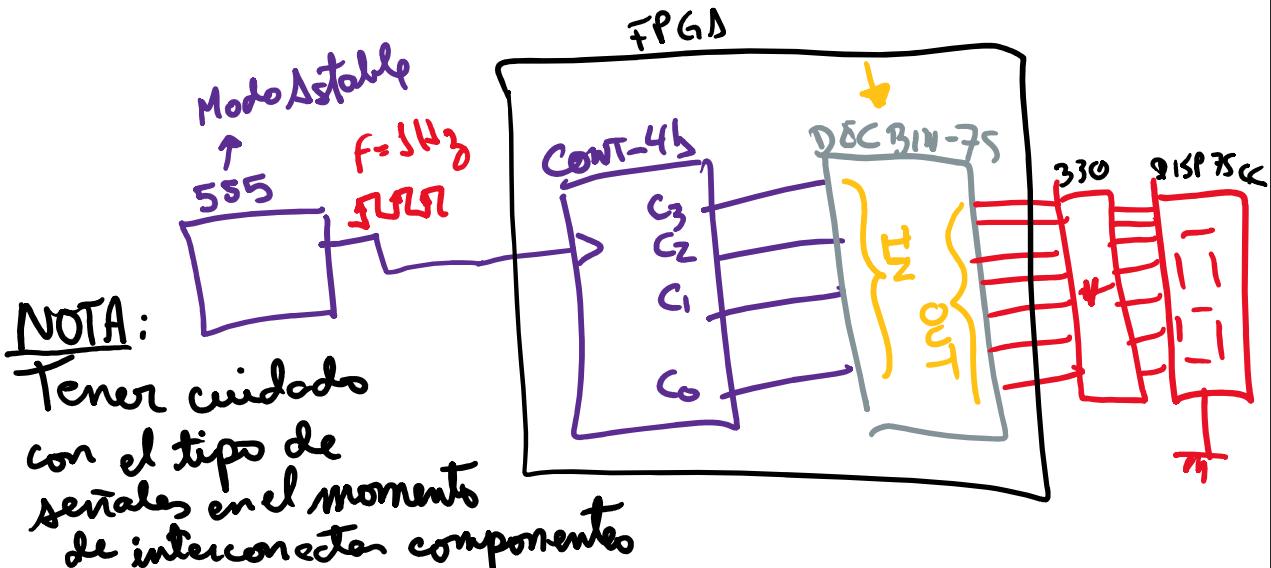
entity DIV_FREQ is
    port(in_clk: in std_logic;
         out_clk: out std_logic);
end DIV_FREQ;

architecture PEPINO of DIV_FREQ is
signal regisron: std_logic_vector(25 downto 0);
signal pre_out_clk: std_logic;
begin
    process(in_clk)
    begin
        if rising_edge(in_clk) then
            regisron <= regisron + 1;
            if regisron = 25000000 then
                regisron <= (others => '0');
                pre_out_clk <= not pre_out_clk;
            end if;
        end if;
    end process;
    out_clk <= pre_out_clk;
end PEPINO;

```

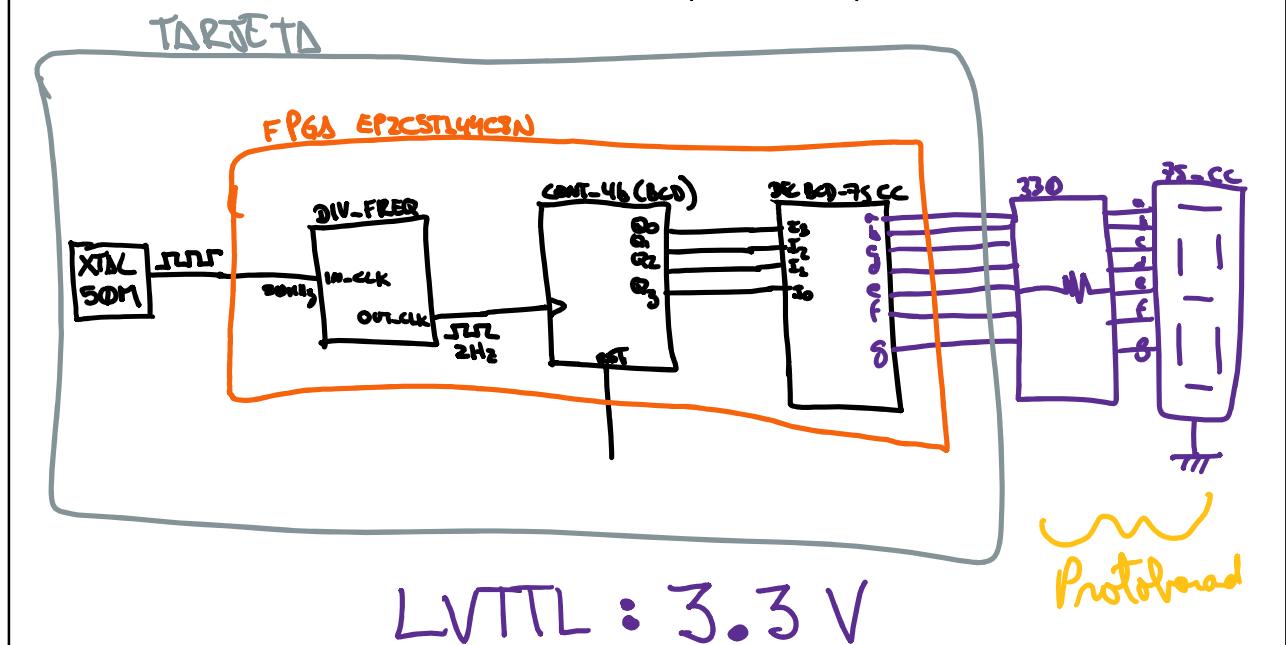
24

Aplicación típica en circuitos lógicos digitales:



25

Circuito de prueba para el laboratorio



26

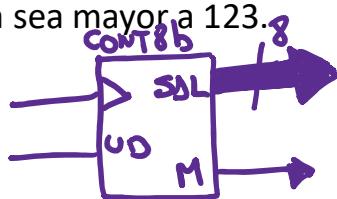
Descripción estructural (RTL) en VHDL

- Pendiente

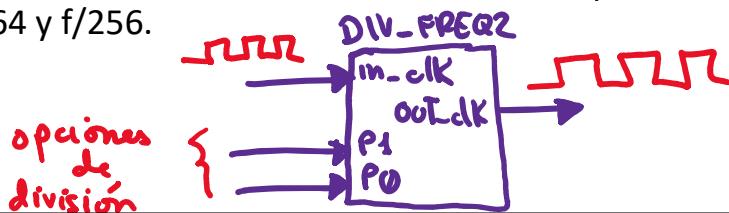
27

Cuestionario:

- Desarrollar un contador binario de 8 bits con opción de cuenta ascendente/descente a través de un pin UD (U=0/D=1). Tener en cuenta una señal de salida adicional de nombre 'M' para que sea '1' cuando la cuenta sea mayor a 123.



- Desarrollar un divisor de frecuencia con opciones de división f/2, f/8, f/64 y f/256.



28

Fin de la sesión de teoría