Universidad de Alcalá

Departamento de Electrónica

Modelado de Sistemas Computacionales

Grado en Ingeniería de Computadores

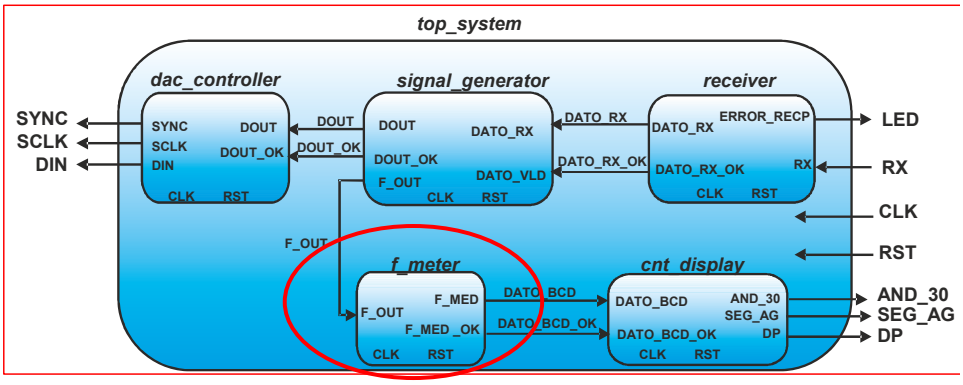
Práctica libre: Generador de señal controlado desde un puerto serie RS232

Apartado 2.

Entidad *f_meter*.

- Entidad *f_meter*..

Es la encargada de obtener el valor de la frecuencia de la señal generada.

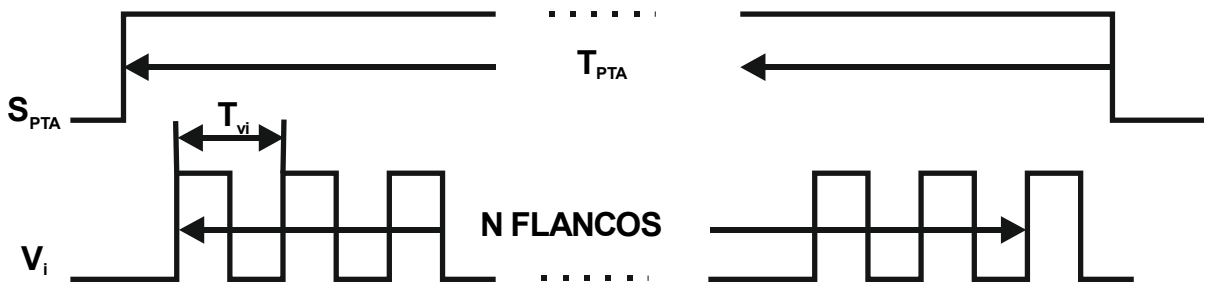


En la salida ***F_MED*** se obtendrá el valor de la frecuencia de ***F_OUT*** codificada en BCD de cuatro dígitos y con una resolución de 1 Hz

Los valores de la frecuencia de ***F_OUT*** estarán comprendidos entre 6 kHz y 90 Hz, aproximadamente.

```
entity f_meter is
  port(
    CLK       : in  std_logic;
    RST       : in  std_logic;
    F_OUT     : in  std_logic;
    F_MED_OK  : out std_logic;
    F_MED     : out std_logic_vector(15 downto 0));
end f_meter;
```

Procedimiento de medida.



$$T_{Vi} = \frac{T_{PTA}}{N} \qquad \Rightarrow f_{Vi} = \frac{N}{T_{PTA}} = N \cdot f_{PTA}$$

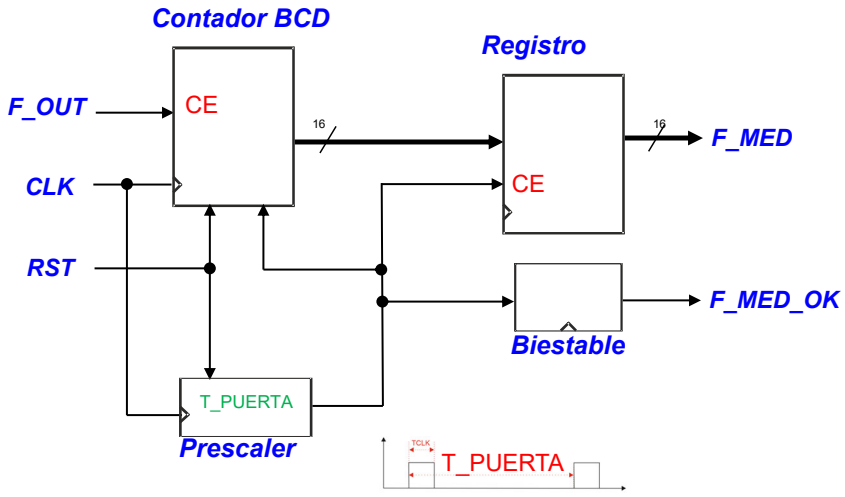
$$T_{PTA} = \begin{cases} 1\text{ s, para implementación} \\ 1\text{ ms, para simulación} \end{cases}$$

- Diagrama de bloques.

Paso 1: Contar.

Paso 2: Medir tiempos.

Paso 3: Registrar.



Entidad *f_meter*

Test-bench.

```
entity f_meter_tb is
end f_meter_tb;

architecture sim of f_meter_tb is
. . . . . .
begin  -- sim

  DUT : entity work.f_meter
    port map (
      CLK      => CLK_i,
      RST      => RST_i,
      F_OUT    => F_OUT_i,
      F_MED_OK => F_MED_OK_i,
      F_MED    => F_MED_i);
. . . . . .
  process
  begin
    F_OUT_i <= '0';
    wait for T-10 ns;
    F_OUT_i <= '1';
    wait for 10 ns;
  end process;

  F <= (1 ms)/T;

  process
  begin  -- pnrocess
. . . . . .
    T <=  ;  -- Poner valor
    wait for 2 ms;

    report "FIN CONTROLADO DE LA SIMULACIÓN" severity failure;

  end process;

end sim;
```

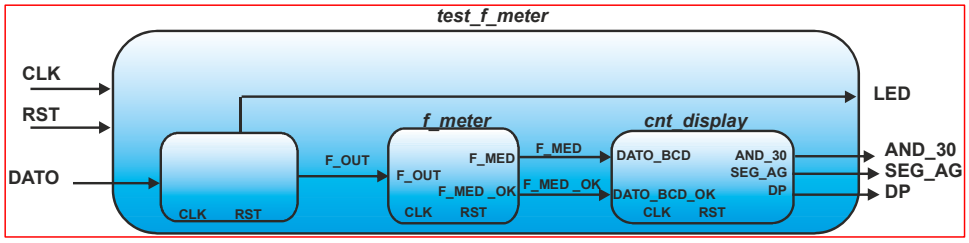
¿Qué hace este proceso?

¿Qué hace esta línea?

¡Hay que completar!

Entidad *f_meter*

Verificación de la entidad *f_meter*..



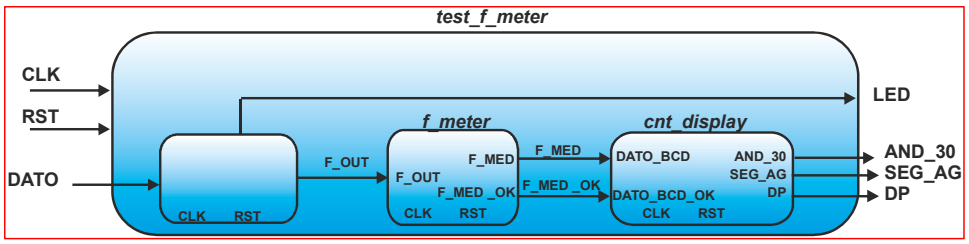

```
entity test_f_meter is
  port (
    CLK     : in  std_logic;
    RST     : in  std_logic;
    DATO    : in  std_logic_vector(12 downto 0);
    LED     : out std_logic_vector(12 downto 0);
    AND_30  : out std_logic_vector(3 downto 0);
    DP      : out std_logic;
    SEG_AG  : out std_logic_vector(6 downto 0));
end test_f_meter;
```

```
architecture rtl of test_f_meter is

  constant CTE       : natural := 10010;
  signal   CNT       : integer range 0 to CTE-1;
  signal   CNT_PROG  : unsigned(12 downto 0);
  signal   ce        : std_logic;
  signal   F_OUT     : std_logic;
  signal   F_MED     : std_logic_vector(15 downto 0);

begin  -- rtl
```

```
U1 : entity work.f_meter
    port map (
      CLK       => CLK,
      RST       => RST,
      F_OUT     => F_OUT,
      F_MED_OK  => F_MED_OK,
      F_MED     => F_MED);

  U2 : entity work.cnt_display
    port map (
      CLK          => CLK,
      RST          => RST,
      DATO_BCD     => F_MED,
      DATO_BCD_OK  => F_MED_OK,
      AND_30       => AND_30,
      DP           => DP,
      SEG_AG       => SEG_AG);
```

Entidad *f_meter*

❑ Verificación de la entidad *f_meter*..




```
LED <= dato;

  process (all)
  begin
    if RST = '1' then
      CNT   <= 0;
      ce    <= '0';
    elsif CLK'event and CLK = '1' then
      if CNT = CTE-1 then
        CNT <= 0;
        ce  <= '1';
      else
        CNT <= CNT+1;
        ce  <= '0';
      end if;
    end if;
  end process;
```

```
process (all)
 begin
   if RST = '1' then
     CNT_PROG     <= (others => '0');
     F_OUT        <= '0';
   elsif CLK'event and CLK = '1' then
     F_OUT        <= '0';
     if ce = '1' then
       if CNT_PROG = unsigned(DATO)-1 then
         CNT_PROG <= (others => '0');
         F_OUT    <= '1';
       else
         CNT_PROG <= CNT_PROG +1;
       end if;
     end if;
   end if;
 end process;
```

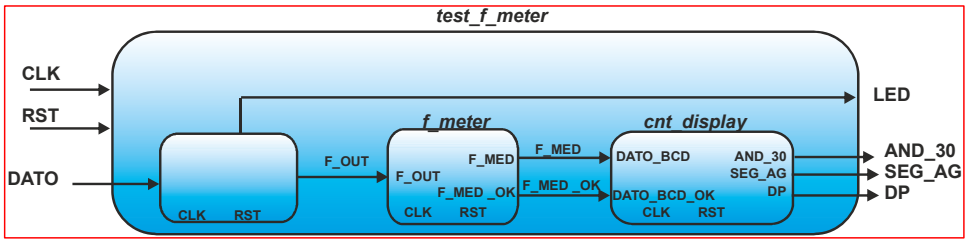
¡Se debe analizar el código para obtener la relación entre el valor de ***DATO*** y la frecuencia visualizada!

¡Se puede preguntar en el examen!

Práctica libre: Generador de señal controlado por el puerto serie RS232

Entidad *f_meter*

❑ Verificación de la entidad *f_meter*..



Esta descarga es imprescindible para la evaluación del apartado. En el caso de que no funcione, se considerará el apartado como no apto.

```
 clock
set_property PACKAGE_PIN W5 [get_ports {CLK}]
set_property IOSTANDARD LVCMOS33 [get_ports {CLK}]

#RST
set_property PACKAGE_PIN T18 [get_ports {RST}]
set_property IOSTANDARD LVCMOS33 [get_ports {RST}]

# Switches
set_property PACKAGE_PIN V17 [get_ports {DATO[0]}]

set_property IOSTANDARD LVCMOS33 [get_ports {DATO[0]}]
set_property PACKAGE_PIN V16 [get_ports {DATO[1]}]

set_property IOSTANDARD LVCMOS33 [get_ports {DATO[1]}]
set_property PACKAGE_PIN W16 [get_ports {DATO[2]}]

set_property IOSTANDARD LVCMOS33 [get_ports {DATO[2]}]
set_property PACKAGE_PIN W17 [get_ports {DATO[3]}]
```