

ASIGNATURA	Modelado y Síntesis de Sistemas Electrónicos Digitales	FECHA	Enero 2019
APELLIDOS, NOMBRE			

Cuestión 1

Dado el siguiente código VHDL

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity cnt2 is
  port (
    RST, LD, PRST, CE : in  std_logic;
    CLK                : in  std_logic;
    Din                : in  std_logic_vector(3 downto 0);
    Q1                 : out std_logic_vector(3 downto 0));
end cnt2;

architecture rtl of cnt2 is
  signal cnt : unsigned(3 downto 0):=(others=>'0');
begin
  process (LD, CLK, CE)
  begin
    if LD = '1' then
      cnt <= unsigned( Din);
    elsif CLK'event and CLK = '1' and CE = '1' then
      cnt <= cnt + 1;
      if RST = '1' then
        cnt <= (others=>'0');
      elsif PRST = '1' then
        cnt <= (3|0=>'1',others=>'0');
      elsif cnt = 9 then
        cnt <= (others=>'0');
      end if;
    end if;
  end process;
  Q1 <= std_logic_vector(cnt);
end rtl;

```

- a- ¿Qué función realiza? Indicar razonadamente el tipo (síncronas/asíncronas) de las entradas

(10 ptos)

- b- Indicar, razonadamente qué errores se cometen en el código anterior si se quiere que sea sintetizable.

(5 ptos)

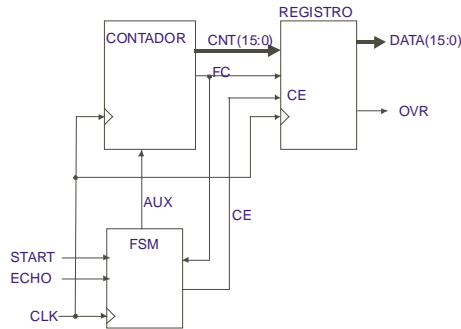
Cuestión 3

Generar el código que permita simular el circuito de la cuestión anterior de forma que la simulación se finalice de forma anticipada una vez que el contador haya pasado 5 veces por el valor $x=45$ y alcance el valor $x=78$.

(15 pts)

Cuestión 4

Se desea diseñar un módulo de medición del tiempo de vuelo ($TOF = Time\ Of\ Flight$) para un sistema de medición de distancias mediante ultrasonidos:



La señal **START** (nivel alto) marca el instante de comienzo de la emisión de la señal ultrasónica. La señal **ECHO** (nivel alto) indica la recepción del eco de la señal ultrasónica, dándose por finalizada la medida. El módulo contador debe contar el número de ciclos de **CLK** comprendidos entre **START** y el flanco ascendente de **ECHO**, obteniéndose así el tiempo transcurrido entre ambos eventos. El número de ciclos se fijará en el bus de salida **DATA** como un número entero sin signo de 16 bits. Las señales **START** y **ECHO** están activas un tiempo igual a un periodo de la señal **CLK**. **CE** se utiliza para actualizar los valores de **DATA** y **OVR**.

El CONTADOR del TOF no puede rebotarse nunca. En el supuesto de que éste alcance el valor máximo (0xFFFF) sin haberse detectado la llegada de un eco, se entregará a la salida data el valor máximo, señalando esta situación de rebose mediante la activación de la salida **OVR**, quedando el sistema a la espera de iniciarse una nueva secuencia de medición activando **START**. La señal **FC** indica, con un nivel alto, que el contado ha alcanzado su valor máximo. **AUX** es una señal que se utiliza para controlar la cuenta del contador.

Con el objeto de obtener una información estable en el bus de salida entre dos medidas consecutivas, las salidas **DATA** y **OVR** serán registradas de forma que su actualización sólo se producirá una vez detectada la recepción del eco (o situación de rebose).

FSM es un sistema secuencial que controla el funcionamiento de CONTADOR y REGISTRO para proporcionar la funcionalidad deseada.

Nota. El sistema dispone de una entrada **RST**, activa a nivel alto, que se utiliza para inicializar todos los bloques secuenciales.

- a- Dibuje el grafo de funcionamiento del sistema secuencial FSM; sabiendo que es de tipo Moore.

(15 ptos)

- a- Crear el código VHDL, sintetizable, que modele el sistema anterior. Se deberá utilizar un proceso para modelar cada uno de los bloques de la figura anterior, con la excepción del bloque FSM para el que se pueden utilizar más de uno.

(30 ptos)

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity C4 is
  port (
    rst    : in  std_logic;
    clk    : in  std_logic;
    start  : in  std_logic;
    echo   : in  std_logic;
    data   : out std_logic_vector(15 downto 0);
    ovr    : out std_logic);
end entity;
```