

**Problema 1 (20 minutos – 2,5 puntos)**

Dadas las cantidades A y B, expresadas en **complemento a dos (Ca2)**:

$$A = 10101010_{Ca2}$$

$$B = 00001111_{Ca2}$$

Se pide:

1. (20%) Convierta a base decimal (dec) con signo A y B:

$$A = \dots\dots\dots \text{dec}$$

$$B = \dots\dots\dots \text{dec}$$

2. (10%) Calcule C en Ca2, siendo:

$$C = A + B = \dots\dots\dots Ca2$$

3. (10%) Calcule D en Ca2, siendo:

$$D = A - B = \dots\dots\dots Ca2$$

4. (10%) En una **suma aritmética binaria con signo**, expresada en **Ca2**, considerando que  $Z = X + Y$ , atendiendo únicamente al bit de signo de X, Y y Z, describa la(s) **condición(es) de Desbordamiento u Overflow (Ov)**:

.....

.....

5. (25%) Complete la **asignación condicionada concurrente** para la **detección de Overflow Ov** (con **Ov = 1**), para el siguiente código VHDL:

```

Library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

Entity SUMA is
port (A,B: in signed(7 downto 0);
      C: out signed(7 downto 0);
      Ov: out std_logic );
End SUMA;

Architecture FUNCIONALIDAD of SUMA is
Signal resultado: signed(7 downto 0);
Begin
  resultado <= A + B;
  C <= resultado;
  Ov <= ..... When ..... Else ..... ;
End FUNCIONALIDAD;
  
```

6. (25%) Analice y complete el siguiente código VHDL para poder calcular en E el complemento a dos (Ca2) de una cantidad D: **E = Ca2 (D). Siendo E y D de 8 bits.**

```

Library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

```

```

Entity COMPDOS is
port (
    D: ..... signed(.....);
    E: ..... signed(.....)
);
End COMPDOS ;

```

```

Architecture FUNCIONALIDAD of COMPDOS is
Begin

```

```

.....

```

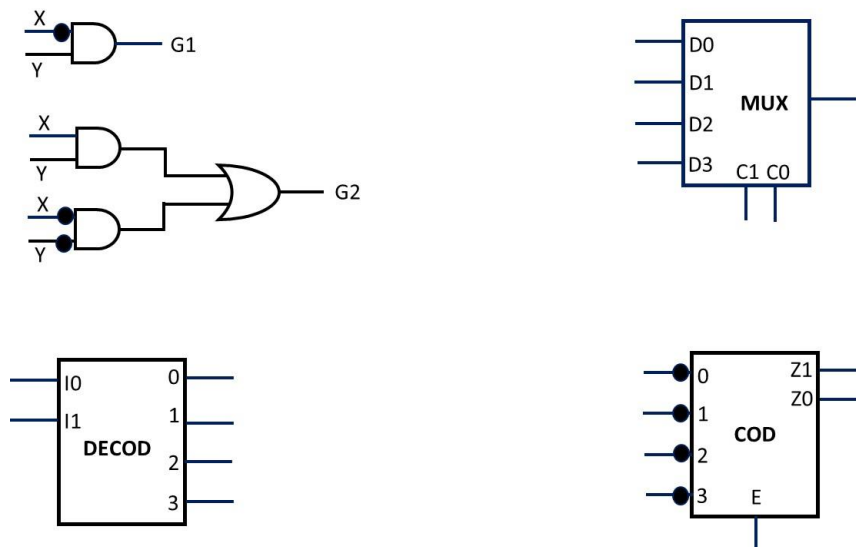
```

End FUNCIONALIDAD;

```

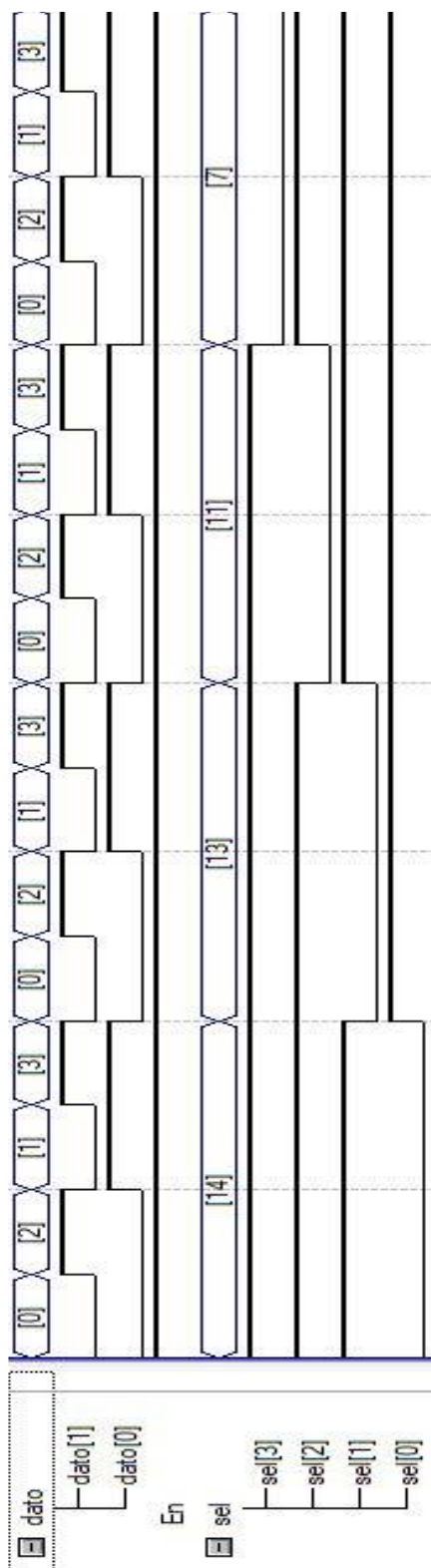
**Problema 2 (30 minutos – 3,5 puntos)**

Un circuito digital contiene los bloques que se muestran en la figura:



**Se pide:**

1. (10%) Tabla de verdad del bloque codificador (COD) y del bloque decodificador (DECOD).
2. (30%) Identifique los bloques en el código VHDL, señale las líneas de código en las que se describe cada uno de los elementos de la figura. Tenga en cuenta que no hay correspondencia de nombres entre la figura y el código.
3. (30%) A partir del código VHDL conecte correctamente los elementos del circuito. Puede unir los elementos por nombre sin necesidad de conectarlos con líneas. Indique en el dibujo los nombres utilizados en el código VHDL.
4. (30%) Complete el siguiente cronograma, añadiendo la señal de salida F del código vhdL.



### CÓDIGO VHDL

```

library ieee;
use ieee.std_logic_1164.all;

```

```

ENTITY circuito IS
  PORT( dato: IN STD_LOGIC_VECTOR(1 DOWNTO 0);
        sel: IN STD_LOGIC_VECTOR(3 DOWNTO 0);
        En: IN STD_LOGIC;
        F: OUT STD_LOGIC);
END circuito;

```

```

ARCHITECTURE func OF circuito IS
  SIGNAL sal: STD_LOGIC_VECTOR(1 DOWNTO 0);
  SIGNAL sal2: STD_LOGIC_VECTOR(3 DOWNTO 0);

```

```

BEGIN

```

```

  process(dato)
  begin
    case dato is
      when "00" => sal2<="0001";
      when "01" => sal2<="0010";
      when "10" => sal2<="0100";
      when "11" => sal2<="1000";
      when others => sal2<="0000";
    end case;
  end process;

  process(sal,dato,sal2)
  begin
    case sal is
      when "00" => F<= (not(dato(1))and dato(0));
      when "01" => F<=((not dato(1))and (not dato(0)))or (dato(1) and
dato(0));
      when "10" => F<=sal2(0);
      when "11" => F<=sal2(3);
      when others => F<='0';
    end case;

```

```
end process;

process(sel,En)
begin
    if En='1' then
        case sel is
            when "1110" => sal<="00";
            when "1101" => sal<="01";
            when "1011" => sal<="10";
            when "0111" => sal<="11";
            when others =>sal<="00";
        end case;
    else
        sal<="00";
    end if;
end process;
END func;
```

**Problema 3 (30 minutos – 4 puntos)**

Se desea diseñar un circuito digital que muestra el nivel de agua de un tanque en números romanos. El circuito leerá la salida de 5 sensores (N5, N4, N3, N2, N1) que se activarán con un uno lógico cuando el agua supere su posición y en caso contrario suministrarán un cero lógico. Para mostrar el nivel de agua detectado por los sensores se utilizan 3 displays que muestran los números romanos de 1 a 5 (I, II, III, IV y V).

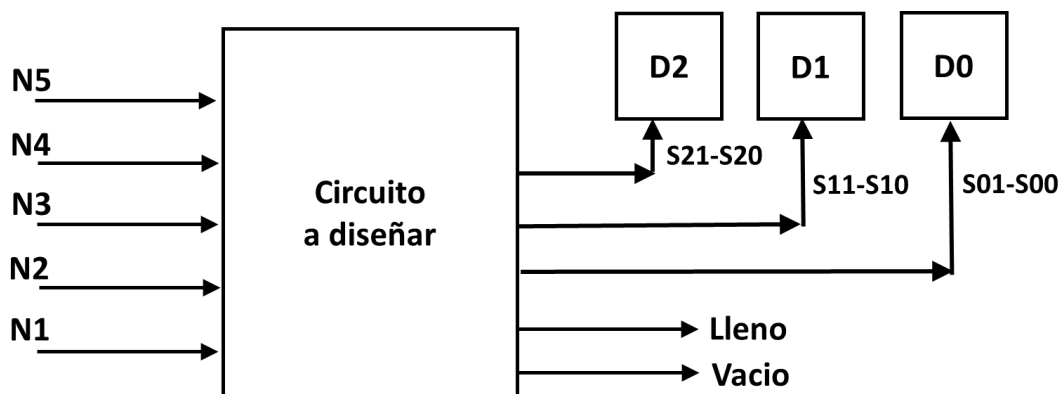
Valores imposibles de los sensores del tanque no mostrarán ningún valor en los displays.

El circuito además activará dos señales de alarma bajo determinadas circunstancias:

- Señal Lleno, cuando el nivel de agua sea máximo. Se activa a nivel alto.
- Señal Vacío, cuando el nivel de agua sea inferior al sensor 1. Se activa a nivel alto.

Cada uno de los 3 displays se controla con dos bits (SX1, SX0, siendo X el número del display utilizado) tal y como se muestra a continuación:

	SX1	SX0
<b>OFF</b>	0	0
<b>I</b>	0	1
<b>V</b>	1	1



Según lo indicado anteriormente se pide:

1. (10%) Complete la siguiente tabla de verdad del sistema completo.

Nivel	N5	N4	N3	N2	N1	S21	S20	S11	S10	S01	S00
<b>I</b>											
<b>II</b>											
<b>III</b>											
<b>IV</b>											
<b>V</b>											

2. (5%) Obtenga la función lógica de la señal lleno.
3. (5%) Obtenga la función lógica de la señal vacío.
4. (10%) Complete la entidad que resulta del sistema digital.

```
ENTITY nivel_R IS
  PORT (
    N5,N4,N3,N2,N1:                ;
    S2: OUT STD_LOGIC_VECTOR( DOWNTO ); --control del display 2
    S1: OUT STD_LOGIC_VECTOR( DOWNTO ); --control del display 1
    S0: OUT STD_LOGIC_VECTOR( DOWNTO ); --control del display 0
  );
END nivel_R;
```

5. (35%) Definir la arquitectura del sistema expuesto haciendo uso de sentencias secuenciales. Indica de forma clara la lista de sensibilidad necesaria.
6. (35%) Definir la arquitectura del sistema expuesto haciendo uso de sentencias concurrentes.