

NOMBRE:

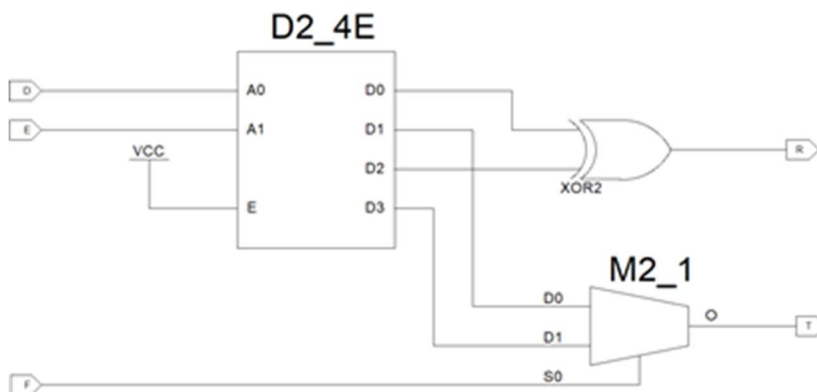
GRUPO:

Observaciones:

- Duración del examen: 1h 40min
- Asegúrese de escribir el nombre y el grupo en todas las hojas

Ejercicio 1.1(2,5 p.)

Dado el siguiente circuito:



- Obtenga la tabla de verdad para las salidas R y T
- Obtenga sus expresiones canónicas
- Realice su diseño en VHDL

a)

b)

D2-4E

DEF	D0	D1	D2	D3	R	T
000	1	0	0	0	1	0
001	1	0	0	0	1	0
010	0	0	1	0	1	0
011	0	0	1	0	1	0
100	0	1	0	0	0	1
101	0	1	0	0	0	0
110	0	0	0	1	0	0
111	0	0	0	1	0	1

$$\begin{aligned}
 R &= \sum_3 (0, 1, 2, 3) = \\
 &= \bar{D}\bar{E}\bar{F} + \bar{D}\bar{E}F + \bar{D}E\bar{F} + \bar{D}EF \\
 &= \prod_3 (4, 5, 6, 7) = \\
 &= (\bar{D} + \bar{E} + F)(\bar{D} + E + \bar{F})(\bar{D} + \bar{E} + F)(\bar{D} + E + \bar{F}) \\
 T &= \sum_3 (4, 7) = D\bar{E}\bar{F} + DEF = \\
 &= \prod_3 (0, 1, 2, 3, 5, 6) = \\
 &= (D + E + F)(D + E + \bar{F})(D + \bar{E} + \bar{F}) \\
 &\quad (D + \bar{E} + \bar{F})(\bar{D} + E + \bar{F})(\bar{D} + E + F)
 \end{aligned}$$

NOMBRE:

GRUPO:

```
c)
library ieee;
use ieee.std_logic_1164.all;
entity ejercicio1 is
    port(D, E, F: in std_logic;
         R, T: out std_logic);
end ejercicio1;
architecture a of ejercicio1 is
    signal A: std_logic_vector(1 downto 0);
    signal S: std_logic_vector(3 downto 0);
begin
    A <= E & D;
    -- Decodificador D2_4E
    process(D, E)
    begin
        case A is
            when "00" => S <= "0001";
            when "01" => S <= "0010";
            when "10" => S <= "0100";
            when others => S <= "1000";
        end case;
    end process;

    R <= S(0) xor S(2);
    T <= S(1) when F = '0' else S(3); -- MUX M2_1
end a;
```

NOMBRE:

GRUPO:

Ejercicio 1.2 (2 p.)

Genere un circuito, con 4 entradas y una salida, que detecte los 5 números menos significativos de su DNI, sin repeticiones. La salida Z se activa cuando el valor binario de las entradas corresponde a alguno de los 5 números. Este circuito debería tener en cuenta también la letra del DNI, de la forma siguiente: si la letra está entre la A y la I, debería detectar el número 11 (0xB), si está entre la J y la Q, debería detectar el número 12 (0xC), y si está entre la R y la Z, debería detectar el número 13 (0xD).

A modo de ejemplo, si su DNI es 40985665R, entonces los números detectados son: 5, 6, 8, 9, 0, y 0xD.

a) Rellene la tabla de verdad

a	b	c	d	Z
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

b) Realice el circuito con un multiplexor de dos entradas de control y las puertas lógicas que considere necesarias

NOMBRE:

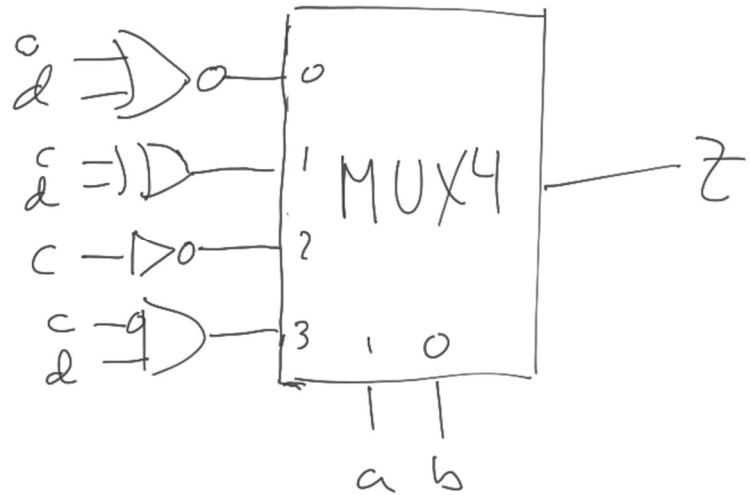
GRUPO:

$$\overline{c} \overline{d} = \overline{c+d}$$

$$c \oplus d$$

$$\overline{c}$$

$$\overline{c} d$$



NOMBRE:

GRUPO:

Ejercicio 1.3 (3 p.)

Dado el siguiente código VHDL que representa un circuito digital combinacional:

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY P2_Parcial IS
    PORT (
        A,B: IN STD_LOGIC_VECTOR ( 1 downto 0) ;
        F0: OUT Std_logic;
        F1: OUT STD_LOGIC_VECTOR (1 downto 0) );
END P2_Parcial;
```

```
ARCHITECTURE funcional OF P2_Parcial IS
    Signal Entrada: STD_LOGIC_VECTOR (3 downto 0);
    Signal AUX1 :STD_LOGIC_VECTOR (1 downto 0);
    Signal AUX2 :STD_LOGIC;

    BEGIN
        AUX1<= (A(0) XNOR B(0))&(B(1));

        Proceso: PROCESS (
            )
        BEGIN
            IF AUX1="10" THEN
                AUX2<=( A(0) NOR B(0) );
            ELSIF AUX1="01" THEN
                AUX2 <= B(0);
            ELSE
                AUX2<=NOT(B(1));
            END IF;
        END PROCESS;

        Entrada<= A&B;

        With Entrada select
            F1<= "00" When "0000"|"0010",
                "01" When "0001"|"0011",
                "10" When "0100"|"0101"|"0110"|"0111",
                "11" When OTHERS;

            F0<=B(0) OR AUX2;
        end funcional;
```

NOMBRE:

GRUPO:

Se pide:

1. (10%) Defina todas las Entradas y Salidas del circuito, indicando el número de bits de cada una de las señales.

Entradas : A y B (2 Bits), 2 Salidas F0(1bit) y F1(2 bits).

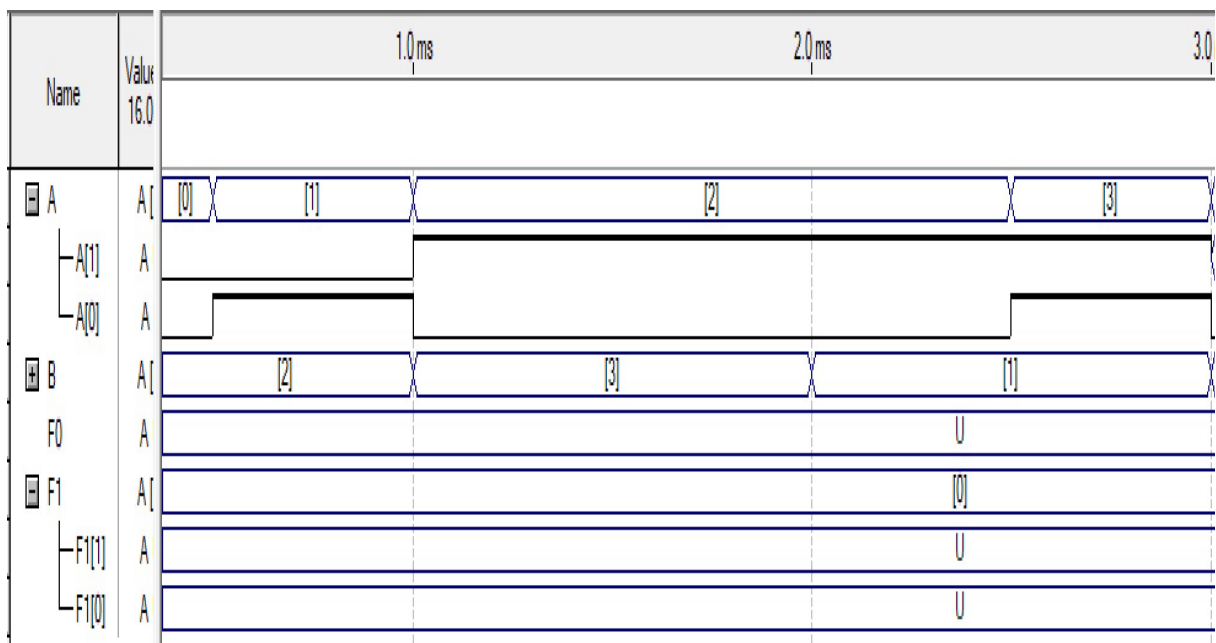
2. (10%) Rellene la lista de sensibilidad del proceso

(AUX1,A(0),B) o (AUX1, A, B)

3. (25%) Complete la siguiente Tabla de verdad correspondiente al Proceso .

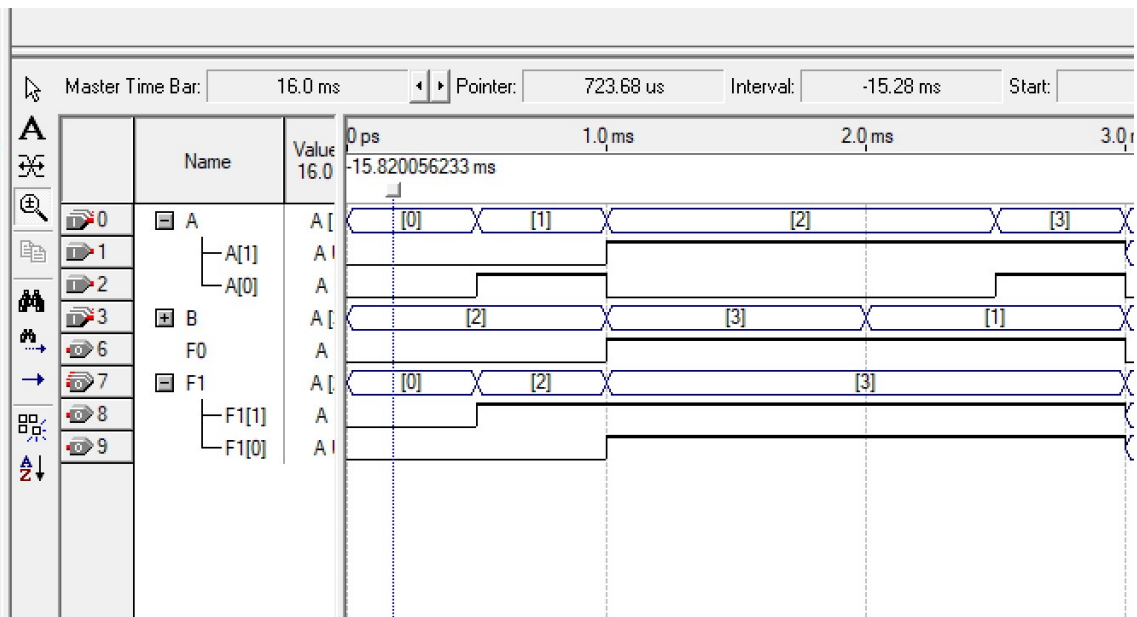
A(0)	B(1)	B(0)	AUX1		AUX2	F0
0	0	0	1	0	1	1
0	0	1	0	0	1	1
0	1	0	1	1	0	0
0	1	1	0	1	1	1
1	0	0	0	0	1	1
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	1	1	1	0	1

1. (25%) Complete el siguiente cronograma de las señales **F0 y F1(0) y F1(1)**, hasta 3 ms.



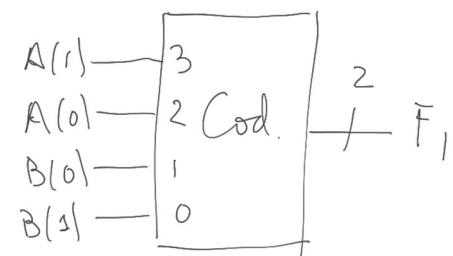
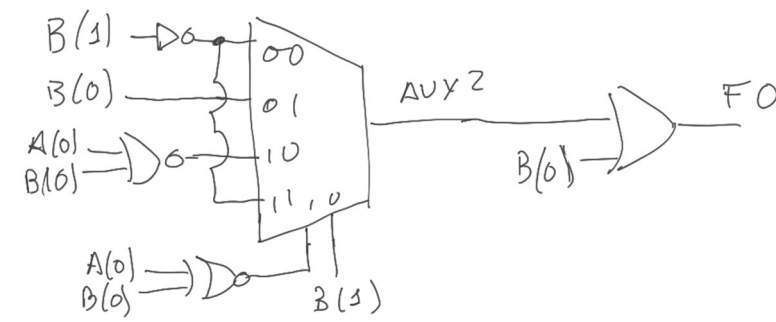
NOMBRE:

GRUPO:



2. (30%) Dibuje el esquema del circuito, que incluya los siguientes Bloques Combinacionales:

Un Multiplexor, un Codificador y las Puertas Lógicas necesarias. **Indique los pesos** de las entradas y salidas de cada bloque.



NOMBRE:

GRUPO:

Ejercicio 1.4 (2,5 p.)

Dados los siguientes **números representados en C_{A2}** , $A=0110100$ y $B=1011010$,

1. Determinar los valores de A y B en decimal. (30%)
2. Llevar a cabo la operación $A-B$ con 7 bits. ¿Se produce desbordamiento? Justifique la respuesta. (40%)
3. Representar B en sistema octal, hexadecimal y BCD. **Para este apartado, asuma que B es un número sin signo.** (30%)

Para los apartados anteriores es necesario indicar el desarrollo, no es suficiente con poner el resultado.

SOLUCIÓN

a) $A = 52$ $B = -38$

b) $A - B = A + (-B) = 0110100 + 0100110 = 1011010$

Se suman dos números positivos y se obtiene un número negativo: Resultado Incorrecto => Overflow

Con complemento a 2 se pueden representar números en el intervalo $-2^{n-1} < \rightarrow 2^{n-1} - 1$ $[-64, 63]$

c) 132_8 ; $5A_{16}$; 10010000_{BCD}