Soluciones Tema 2

Ejercicio 1.

$$Y = \left(\overline{A + B}\right)\overline{C} + \overline{C} \cdot D$$

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio1 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio1;

architecture arquitectura1 of Ejercicio1 is

begin

y<='0' when (C='1') else (A nor B) or D;

end arquitectura1;
```

Otra posibilidad

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity Ejercicio1 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio1;

architecture arquitectura2 of Ejercicio1 is

begin

y<= '1' when (A&B&C&D="000-")else
    '1' when (A&B&C&D="0101")else
    '1' when (A&B&C&D="1001")else
    '1' when (A&B&C&D="1101")else
    '0';

end arquitectura2;
```

Otra posibilidad

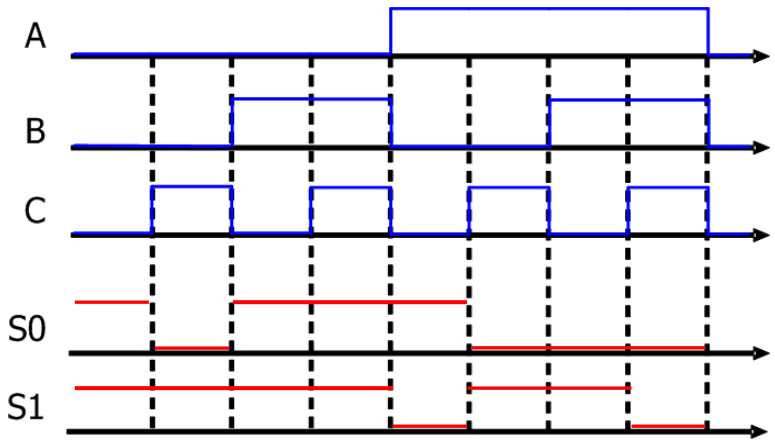
```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio1 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio1;

architecture arquitectura2 of Ejercicio1 is

begin

y<= not C when (D='1') else not(A or B or C);

end arquitectura2;
```

Otra posibilidad

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio1 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio1;

architecture arquitectura2 of Ejercicio1 is

begin

y<= '1' when ((A='0' and B='0' and C='0')or (C='0' and D='1') else '0';

end arquitectura2;
```

Ejercicio 2.

(a)



(b)

$$S0 = \overline{B}\cdot\overline{C} + \overline{A}\cdot B$$

$$S1 = \overline{A} + \overline{B}\cdot C + B\cdot\overline{C}$$

(c)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio2 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           S0 : out  STD_LOGIC;
           S1 : out  STD_LOGIC);
end Ejercicio2;

architecture arquitectura1 of Ejercicio2 is

begin

 S0<= (B nor C) when (A='1') else (B or (not c));
 S1<= '1' when (A='0') else B xor C;

end arquitectura1;
```

Ejercicio 3.

(a)

$$Y = A\cdot\overline{B}\cdot\overline{D} + B\cdot D$$

$$\mathrm{Y} = \left(\mathrm{A} + \mathrm{B}\right)\cdot\left(\overline{\mathrm{B}} + \mathrm{D}\right)\cdot\left(\overline{\mathrm{A}} + \overline{\mathrm{D}}\right)$$

$$Y = \left(B + \overline{D}\right)\cdot(\,\overline{B} + D\,)\cdot(\,A + D\,) = \left(A + B\right)\cdot\left(\overline{B \oplus D}\right)$$

$$Y = \left(B + \overline{D}\right)\left(A + D\right)\cdot\left(\overline{A} + \overline{B}\right)$$

$$Y = \left(A + B\right)\left(\overline{B} + D\right)\left(B + \overline{D}\right) = \left(A + B\right)\cdot\left(\overline{B \oplus D}\right)$$

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity Ejercicio3_1 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio3_1;

architecture arquitectura1 of Ejercicio3_1 is

begin

 y<= (B and D) or (A and (not B) and (not D));

end arquitectura1;
```

(b)

A partir del mapa de karnaugh

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio3_2_2 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio3_2_2;

architecture arquitectura2 of Ejercicio3_2_2 is

begin
 Y<= '1' when (A='0' and B='1' and C='1' and D='1') or (A='1' and B='0' and
 C='1' and D='0') else
'-' when (C='0' and (A='1' or B='1' or D='1')) or (A='1' and B='1' and
C='1' and D= '1') else
        '0';

end arquitectura2;
```

Otra posibilidad a partir del mapa de karnaugh

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity Ejercicio3_2_2 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio3_2_2;

architecture arquitectura2 of Ejercicio3_2_2 is

signal entrada : std_logic_vector (3 downto 0);

begin
 entrada<=A&B&C&D;

 Y<= '1' when (entrada="0111" or entrada="1010") else
     '0' when (entrada="0000" or entrada="001-" or entrada="-110" or
        entrada="1011") else
     '-';
end arquitectura2;
```

Otra posibilidad a partir de la primera ecuación del apartado a)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity Ejercicio3_2 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio3_2;

architecture arquitectura2 of Ejercicio3_2 is

begin

 Y<= (A  and (not D)) when (B='0') else D;


end arquitectura2;
```

Otra posibilidad sería a partir de la primera ecuación del apartado a)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity Ejercicio3_2 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio3_2;

architecture arquitectura2 of Ejercicio3_2 is

begin

 Y<= B when (D='1') else (A and (not B);

end arquitectura2;
```

(c)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity Ejercicio3_3 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio3_3;

architecture arquitectura3 of Ejercicio3_3 is
 signal entrada : std_logic_vector (3 downto 0);
begin
 entrada<=A&B&C&D;
 with entrada select

    Y<='0' when "0000",
       '0' when "001-",
       '0' when "0110",
       '0' when "1110",
       '0' when "1011",
       '1' when "0111",
       '1' when "1010",
       '-' when others;

end arquitectura3;
```

Otra posibilidad

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity Ejercicio3_3 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           D : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejercicio3_3;

architecture arquitectura3 of Ejercicio3_3 is
 signal entrada : std_logic_vector (3 downto 0);
begin
 entrada<=A&B&C&D;
 with entrada select

    Y<='0' when "0000"|"001-"|"0110"|"1110"|"1011",
       '1' when "0111"|"1010",
       '-' when others;

end arquitectura3;
```

Ejercicio 4.

(a)

$$Y = B \cdot C + A \cdot C + A \cdot B$$

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejemplo4 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejemplo4;

architecture arquitectura3 of Ejemplo4 is

begin

  y<= (B and C) or (A and C) or (A and B);

end arquitectura3;
```

(b)

```
library IEEE;

use IEEE.STD_LOGIC_1164.ALL;

entity Ejemplo4 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejemplo4;

architecture arquitectura1_1 of Ejemplo4 is

begin

  y<= B or C when (A='1') else B and C;

end arquitectura1_1;
```

Lo mismo de antes se puede hacer con las otras dos variables, es decir:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejemplo4 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejemplo4;


architecture arquitectura1_1 of Ejemplo4 is

begin

 y<= A or C when (B='1') else A and C;

end arquitectura1_1;


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejemplo4 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejemplo4;

architecture arquitectura1_1 of Ejemplo4 is

begin

 y<=A or B when (C='1') else A and B;

end arquitectura1_1;
```

(c)

```
library IEEE;

use IEEE.STD_LOGIC_1164.ALL;

entity Ejemplo4 is
    Port ( A : in  STD_LOGIC;
           B : in  STD_LOGIC;
           C : in  STD_LOGIC;
           Y : out  STD_LOGIC);
end Ejemplo4;

architecture arquitectura1 of Ejemplo4 is

 signal entrada : std_logic_vector (2 downto 0);

begin
   entrada<=A&B&C;
 with entrada select

  y<= '1' when "11-",
      '1' when "011",
      '1' when "101",
      '0' when others;

end arquitectura1;
```

Ejercicio 5.(a)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity Ejercicio5_1 is
    Port ( M1 : in  STD_LOGIC;
           M2 : in  STD_LOGIC;
           M3 : in  STD_LOGIC;
           M4 : in  STD_LOGIC;
           TA : out  STD_LOGIC;
           MA : out  STD_LOGIC;
           I : out  STD_LOGIC;
           MR : out  STD_LOGIC);
end Ejercicio5_1;

architecture arquitectura1 of Ejercicio5_1 is

begin
 TA <= '1' when (M1&M2&M3&M4="1111") else '0';
 MA <= '1' when (M1&M2&M3&M4="0111" or M1&M2&M3&M4="1011" or
       M1&M2&M3&M4="1101" or M1&M2&M3&M4="1110" else
       '0';
 I <= '1' when (M1&M2&M3&M4="0011" or M1&M2&M3&M4="0101" or
      M1&M2&M3&M4="0110" or M1&M2&M3&M4="1001" or M1&M2&M3&M4="1010" or
      M1&M2&M3&M4="1100") else
      '0';
 MR <='1'when (M1&M2&M3&M4="000-" or M1&M2&M3&M4="0010" or
      M1&M2&M3&M4="0100" or M1&M2&M3&M4="1000" else
'0';

end arquitectura1;
```

Otra posibilidad

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio5_1 is
    Port ( M1 : in  STD_LOGIC;
           M2 : in  STD_LOGIC;
           M3 : in  STD_LOGIC;
           M4 : in  STD_LOGIC;
           TA : out  STD_LOGIC;
           MA : out  STD_LOGIC;
           I : out  STD_LOGIC;
           MR : out  STD_LOGIC);
end Ejercicio5_1;

architecture arquitectura1 of Ejercicio5_1 is

begin

TA <='1' when (M1='1' and M2='1' and M3='1' and M4='1') else
     '0';
MA <='1' when (M1='0' and M2='1' and M3='1' and M4='1') else
     '1' when (M1='1' and M2='0' and M3='1' and M4='1') else
     '1' when (M1='1' and M2='1' and M3='0' and M4='1') else
     '1' when (M1='1' and M2='1' and M3='1' and M4='0') else
     '0';
I <= '1' when (M1='0' and M2='0' and M3='1' and M4='1') else
     '1' when (M1='0' and M2='1' and M3='0' and M4='1') else
     '1' when (M1='0' and M2='1' and M3='1' and M4='0') else
     '1' when (M1='1' and M2='0' and M3='0' and M4='1') else
     '1' when (M1='1' and M2='0' and M3='1' and M4='0') else
     '1' when (M1='1' and M2='1' and M3='0' and M4='0') else
     '0';
 MR<='1' when (M1='0' and M2='0' and M3='0') else
     '1' when (M1='0' and M2='0' and M4='0') else
     '1' when (M1='0' and M3='0' and M4='0') else
     '1' when (M2='0' and M3='0' and M4='0') else
     '0';
end arquitectura1;
```

(b)

```
library IEEE;

use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio5_2 is
    Port ( M1 : in  STD_LOGIC;
           M2 : in  STD_LOGIC;
           M3 : in  STD_LOGIC;
           M4 : in  STD_LOGIC;
           TA : out  STD_LOGIC;
           MA : out  STD_LOGIC;
           I : out  STD_LOGIC;
           MR : out  STD_LOGIC);
end Ejercicio5_2;
```

```
architecture arquitectura2 of Ejercicio5_2 is
 signal entradas : std_logic_vector (3 downto 0);
 signal salidas : std_logic_vector (3 downto 0);

begin

 entradas<=M1&M2&M3&M4;
 with entradas select
       salidas <= "0001" when "000-",
                  "0001" when "0010",
                   "0001" when "0100",
                   "0001" when "1000",
                   "0100" when "0111",
                   "0100" when "1011",
                   "0100" when "1101",
                   "0100" when "1110",
                   "1000" when "1111",
                   "0010" when others;


       TA<=salidas(3);
       MA<=salidas(2);
       I<=salidas(1);
       MR<=salidas(0);

end arquitectura2;
```

Otra posibilidad

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio5_3 is
    Port ( entradas : in std_logic_vector (3 downto 0);
           salidas : out std_logic_vector (3 downto 0));
end Ejercicio5_3;

architecture arquitectura3 of Ejercicio5_3 is

begin

 with entradas select
       salidas <= "0001" when "000-",
                  "0001" when "0010",
                   "0001" when "0100",
                   "0001" when "1000",
                   "0100" when "0111",
                   "0100" when "1011",
                   "0100" when "1101",
                   "0100" when "1110",
                   "1000" when "1111",
                   "0010" when others;

end arquitectura3;
```

Otra posibilidad

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio5_3 is
    Port ( entradas : in std_logic_vector (3 downto 0);
           salidas : out std_logic_vector (3 downto 0));
end Ejercicio5_3;

architecture arquitectura3 of Ejercicio5_3 is

begin

 with entradas select
       salidas <= "0001" when "000-"|"0010"|"0100"|"1000",
                  "0100" when "0111"|"1011"|"1101"|"1110",
                  "1000" when "1111",
                  "0010" when others;

end arquitectura3;
```

Otra posibilidad

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Ejercicio5_4 is
    Port ( M1 : in  STD_LOGIC;
           M2 : in  STD_LOGIC;
           M3 : in  STD_LOGIC;
           M4 : in  STD_LOGIC;
           TA : out  STD_LOGIC;
           MA : out  STD_LOGIC;
           I : out  STD_LOGIC;
           MR : out  STD_LOGIC);
end Ejercicio5_4;

architecture arquitectura4 of Ejercicio5_4 is
 signal entradas : std_logic_vector (3 downto 0);
begin

 entradas<=M1&M2&M3&M4;


 with entradas select
       TA <= '1' when "1111",
             '0' when others;

 with entradas select
       MA <= '1' when "0111",
             '1' when "1011",
             '1' when "1101",
             '1' when "1110",
             '0' when others;
```

```
 with entradas select
        I <= '1' when "0011",
             '1' when "0101",
             '1' when "0110",
             '1' when "1001",
             '1' when "1010",
             '1' when "1100",
             '0' when others;

 with entradas select
       MR <= '1' when "0000",
             '1' when "0001",
             '1' when "0010",
             '1' when "0100",
             '0' when others;


end arquitectura4;
```