

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERU
FACULTAD DE ESTUDIOS GENERALES CIENCIAS



Solución a las experiencias del Laboratorio 4

EXPERIENCIA 1 Y EXPERIENCIA 2 DEL LABORATORIO 4

ALUMNO:

MATEO GUERRERO ISUIZA

CÓDIGO:

20191867

HORARIO:

0441

PROFESOR:

Flores Espinoza Donato Andrés

2020-2

Lima, Octubre, 2020

1. EXPERIENCIA 1

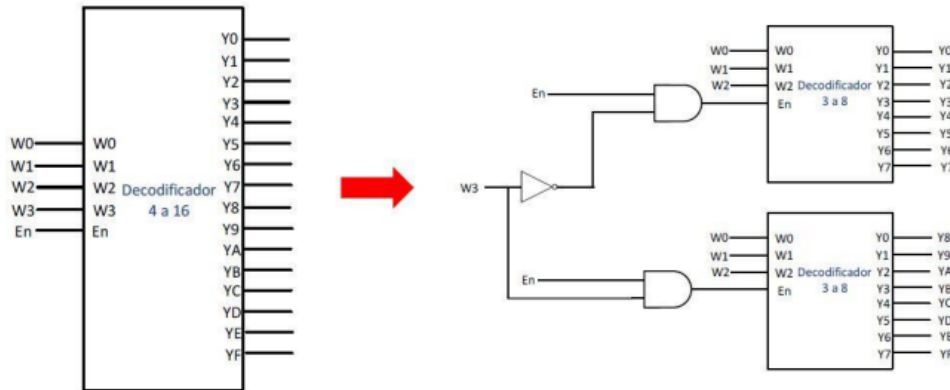
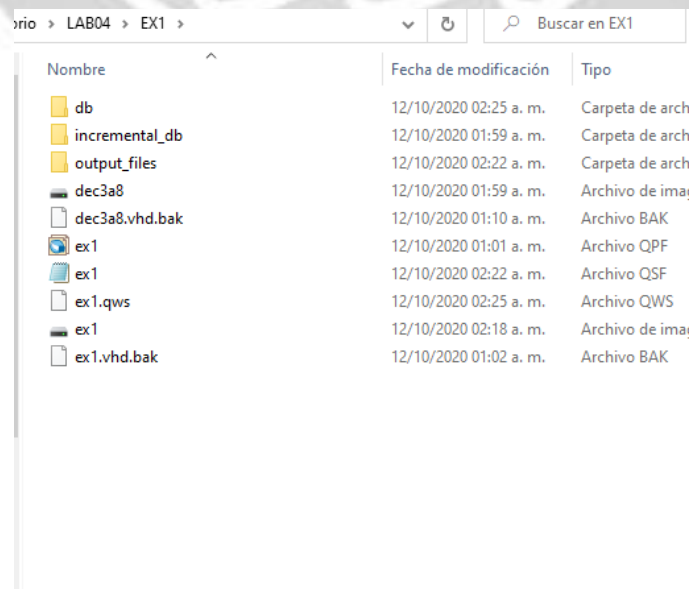


Fig 1 - dec4a16 [1]

Tabla 1 – Representa f(a,b,c,d)

Nº FILA	A	B	C	D	F
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	d
9	1	0	0	1	d
10	1	0	1	0	d
11	1	0	1	1	d
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

- i. Cree el proyecto ex1 (circuito principal ex1.vhd).



- ii. Realice el circuito dec3a8 (archivo dec3a8.vhd), luego añádalo al proyecto

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  ENTITY dec3a8 IS
5      PORT ( EN      : IN  STD_LOGIC;
6            W       : IN  STD_LOGIC_VECTOR(2 DOWNT0 0);
7            Y       : OUT STD_LOGIC_VECTOR(7 DOWNT0 0));
8  END dec3a8;
9
10 ARCHITECTURE logic1 OF dec3a8 IS
11     SIGNAL X: STD_LOGIC_VECTOR(3 DOWNT0 0);
12 BEGIN
13     X<= EN&W;
14     Y<= "00000001" when X="1000" ELSE
15         "00000100" when X="1001" ELSE
16         "00001000" when X="1010" ELSE
17         "00001000" when X="1011" ELSE
18         "00010000" when X="1100" ELSE
19         "00100000" when X="1101" ELSE
20         "01000000" when X="1110" ELSE
21         "10000000" when X="1111" ELSE
22         "00000000";
23 END logic1;
24
25
26
27

```

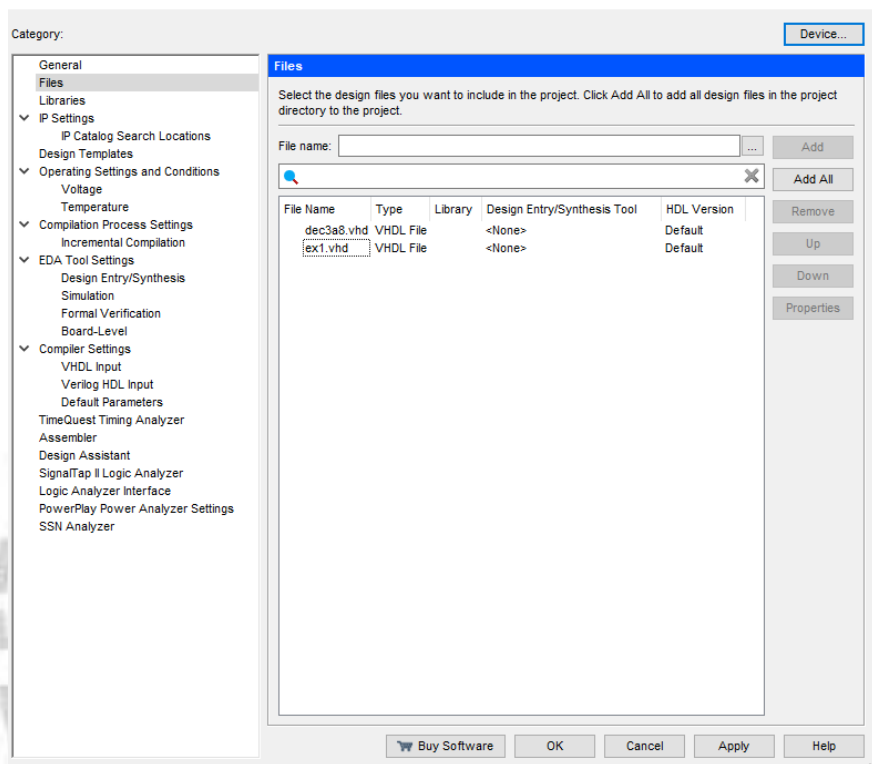
- iii. Realizar la descripción del circuito ex1, (archivo ex1.vhd). El cuál será el dec4a16 adaptado para poder realizar la función lógica indicada. No cree un package. Añada el componente dec3a8 tal como haría en un package, pero realícelo entre la definición de la arquitectura del ex1 y el respectivo begin. Emplee las instrucciones enseñadas en clase para realizar el diseño jerárquico.

```

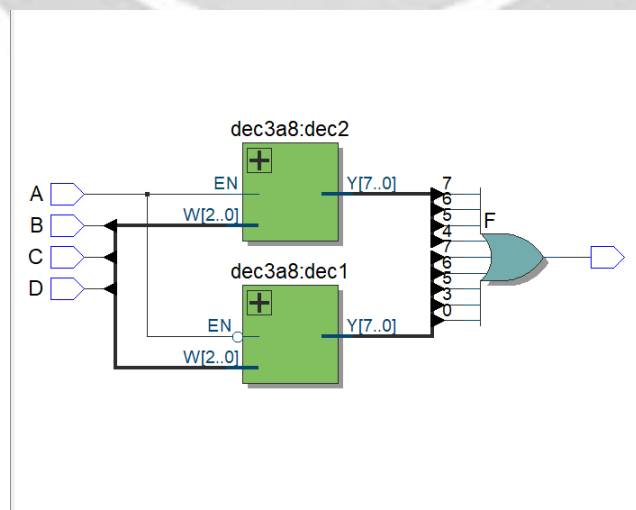
1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  ENTITY ex1 IS
5      PORT ( A,B,C,D : IN  STD_LOGIC;
6            F       : OUT STD_LOGIC);
7  END ex1;
8
9  ARCHITECTURE logic1 OF ex1 IS
10     SIGNAL X: STD_LOGIC_VECTOR(2 DOWNT0 0);
11     SIGNAL Q: STD_LOGIC_VECTOR(15 DOWNT0 0);
12     SIGNAL EN, EN1, EN2 : STD_LOGIC;
13     COMPONENT dec3a8
14         PORT ( EN      : IN  STD_LOGIC;
15               W       : IN  STD_LOGIC_VECTOR(2 DOWNT0 0);
16               Y       : OUT STD_LOGIC_VECTOR(7 DOWNT0 0));
17     END COMPONENT;
18 BEGIN
19     X<= B & C & D;
20     EN<= '1';
21     EN1<= (not(A) and EN);
22     EN2<= (A and EN);
23     dec1: dec3a8 PORT MAP (EN1,X,Q(7 DOWNT0 0));
24     dec2: dec3a8 PORT MAP (EN2,X,Q(15 DOWNT0 8));
25     F<= Q(0) OR Q(3) OR Q(5) OR Q(6) OR Q(7)OR Q(12)
26 END logic1;

```

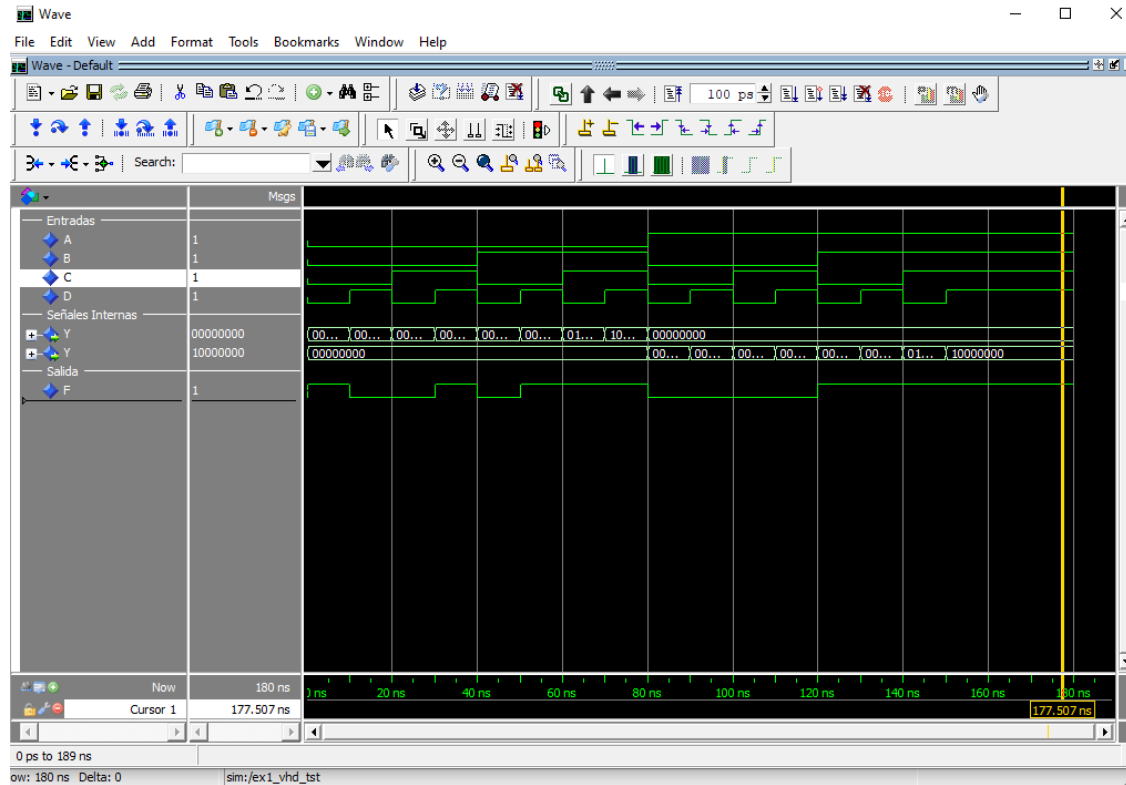
- iv. Aclaración: Antes de compilar (realizar la síntesis) entre en: Project -> add/remove files in Project. Asegúrese de que el orden de los archivos sea: dec3a8.vhd (parte superior), ex1.vhd (parte inferior), (cualquier cosa emplee los botones de Up o Down).



- v. Interpretar el diseño del circuito ex1 por medio de RTL Viewer.



- vi. Realizar la simulación del circuito ex1 (añada las señales internas) y luego automatice por medio del start.do (tal como indica la guía de la sesión N°03)



2. EXPERIENCIA 2

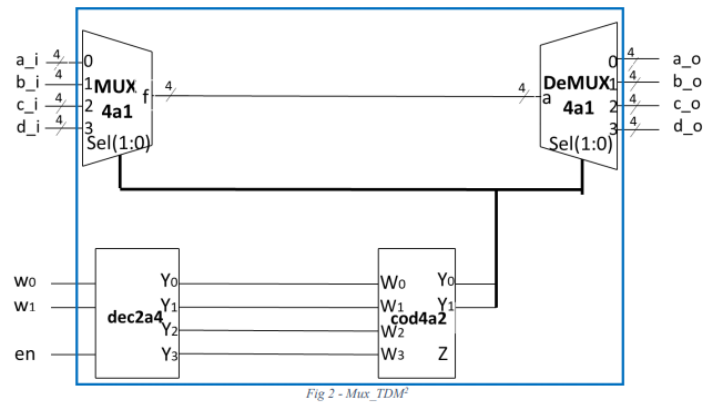
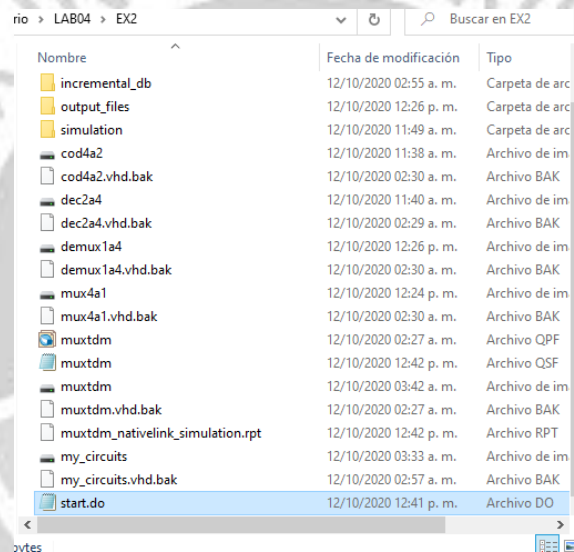


Fig 2 - Mux_TDM

- i. Cree el proyecto muxtdm (circuito principal muxtdm.vhd))



- ii. Realice los siguientes circuitos y añádalos al proyecto (guíese de la Fig 2): a. dec2a4 (archivo dec2a4.vhd b. cod4a2 (archivo cod4a2.vhd) c. mux4a1 (archivo mux4a1.vhd) d. demux1a4(archivo demux1a4.vhd)

Quartus Prime Lite Edition - C:/Users/ANDREA/Desktop/LAB04/EX2/muxtdm - muxtdm

File Edit View Project Assignments Processing Tools Window Help

Project Navigator Hierarchy Entity Instance

MAX 10: 10M50DAF484C7G

- muxtdm
 - cod4a2.co
 - dec2a4.deco
 - demux1a4.demux
 - mux4a1.mux

Tasks/Compilation Task

- Compile Design
- Analysis & Synthesis
- Edit Settings
- View Report
- Analysis & Elaboration
- Partition Merge
- Netlist Viewers

Messages

Type	ID	Message
Information		Running Quartus Prime Netlist Viewers Preprocess
Information		Command: quartus_rpp muxtdm -c muxtdm --netlist_type=sgate
Information		Quartus Prime Netlist Viewers Preprocess was successful. 0 errors, 0 warnings

System (18) Processing (23)

Ln 20 Col 10 VHDL File

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  ENTITY dec2a4 IS
5  PORT (w : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
6       EN : IN STD_LOGIC;
7       Y : OUT STD_LOGIC_VECTOR(3 DOWNTO 0));
8  END dec2a4;
9
10 ARCHITECTURE logic4 OF dec2a4 IS
11   SIGNAL Q : STD_LOGIC_VECTOR(2 DOWNTO 0);
12 BEGIN
13   Q<=EN&w;
14   Y<="0001" WHEN Q="100" ELSE
15     "0010" WHEN Q="101" ELSE
16     "0100" WHEN Q="110" ELSE
17     "1000" WHEN Q="111" ELSE
18     "0000";
19 END logic4;
20

```

Quartus Prime Lite Edition - C:/Users/ANDREA/Desktop/LAB04/EX2/muxtdm - muxtdm

File Edit View Project Assignments Processing Tools Window Help

Project Navigator Hierarchy Entity Instance

MAX 10: 10M50DAF484C7G

- muxtdm
 - cod4a2.co
 - dec2a4.deco
 - demux1a4.demux
 - mux4a1.mux

Tasks/Compilation Task

- Compile Design
- Analysis & Synthesis
- Edit Settings
- View Report
- Analysis & Elaboration
- Partition Merge
- Netlist Viewers

Messages

Type	ID	Message
Information		Running Quartus Prime Netlist Viewers Preprocess
Information		Command: quartus_rpp muxtdm -c muxtdm --netlist_type=sgate
Information		Quartus Prime Netlist Viewers Preprocess was successful. 0 errors, 0 warnings

System (18) Processing (23)

Ln 17 Col 10 VHDL File

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  ENTITY cod4a2 IS
5  PORT (w : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
6       Y : OUT STD_LOGIC_VECTOR(1 DOWNTO 0));
7  END cod4a2;
8
9  ARCHITECTURE logic3 OF cod4a2 IS
10 BEGIN
11   Y<="00" WHEN w="0001" ELSE
12     "01" WHEN w="0010" ELSE
13     "10" WHEN w="0100" ELSE
14     "11" WHEN w="1000" ELSE
15     "--";
16 END logic3;
17

```

Quartus Prime Lite Edition - C:/Users/ANDREA/Desktop/LAB04/EX2/muxtdm - muxtdm

File Edit View Project Assignments Processing Tools Window Help

Project Navigator Hierarchy Entity Instance

MAX 10: 10M50DAF484C7G

- muxtdm
 - cod4a2.co
 - dec2a4.deco
 - demux1a4.demux
 - mux4a1.mux

Tasks/Compilation Task

- Compile Design
- Analysis & Synthesis
- Edit Settings
- View Report
- Analysis & Elaboration
- Partition Merge
- Netlist Viewers

Messages

Type	ID	Message
Information		Running Quartus Prime Netlist Viewers Preprocess
Information		Command: quartus_rpp muxtdm -c muxtdm --netlist_type=sgate
Information		Quartus Prime Netlist Viewers Preprocess was successful. 0 errors, 0 warnings

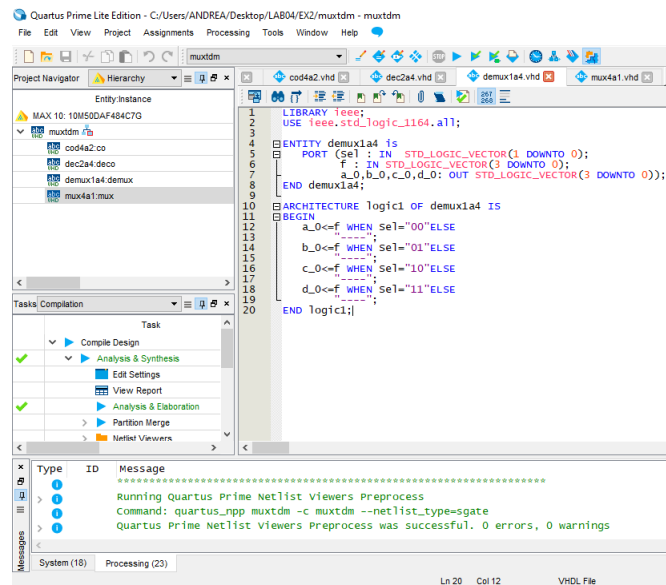
System (18) Processing (23)

Ln 17 Col 12 VHDL File

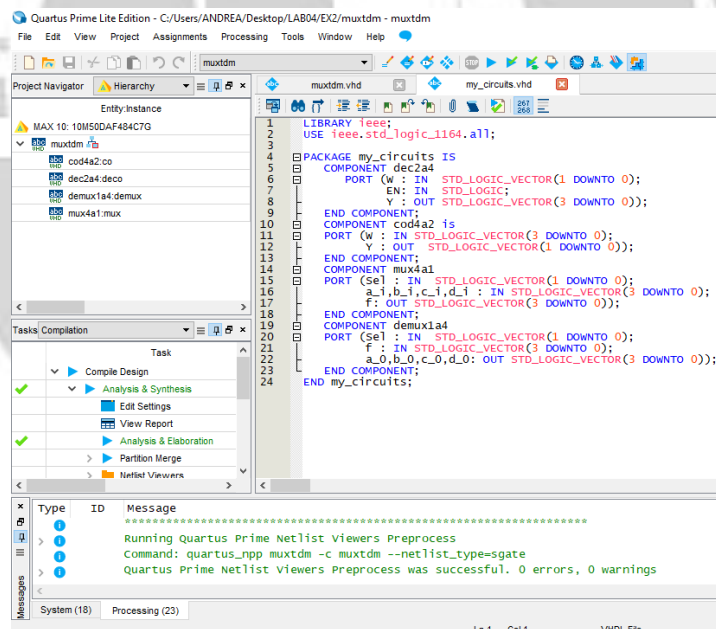
```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  ENTITY mux4a1 IS
5  PORT (sel : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
6       a_1,b_1,c_1,d_1 : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
7       Y : OUT STD_LOGIC_VECTOR(3 DOWNTO 0));
8  END mux4a1;
9
10 ARCHITECTURE logic2 OF mux4a1 IS
11 BEGIN
12   f<=a_1 WHEN sel="00" ELSE
13     b_1 WHEN sel="01" ELSE
14     c_1 WHEN sel="10" ELSE
15     d_1 WHEN sel="11" ELSE
16     "--";
17 END logic2;

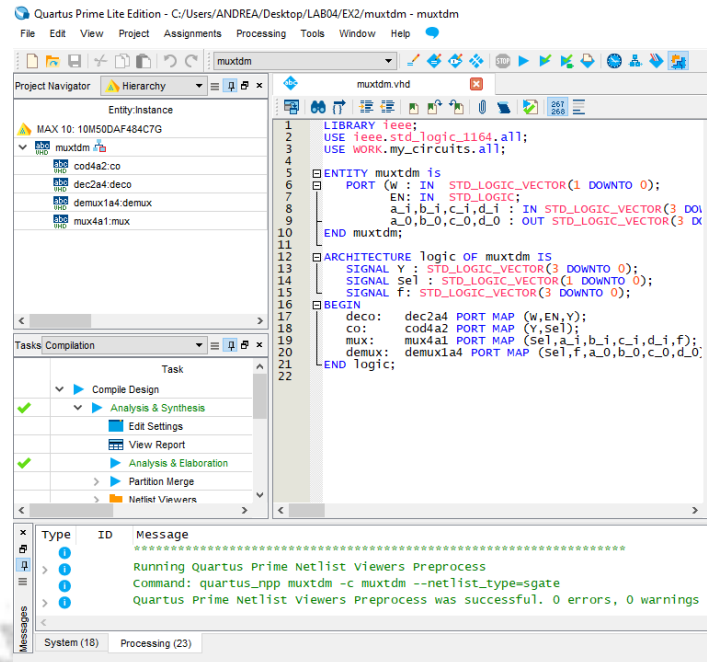
```



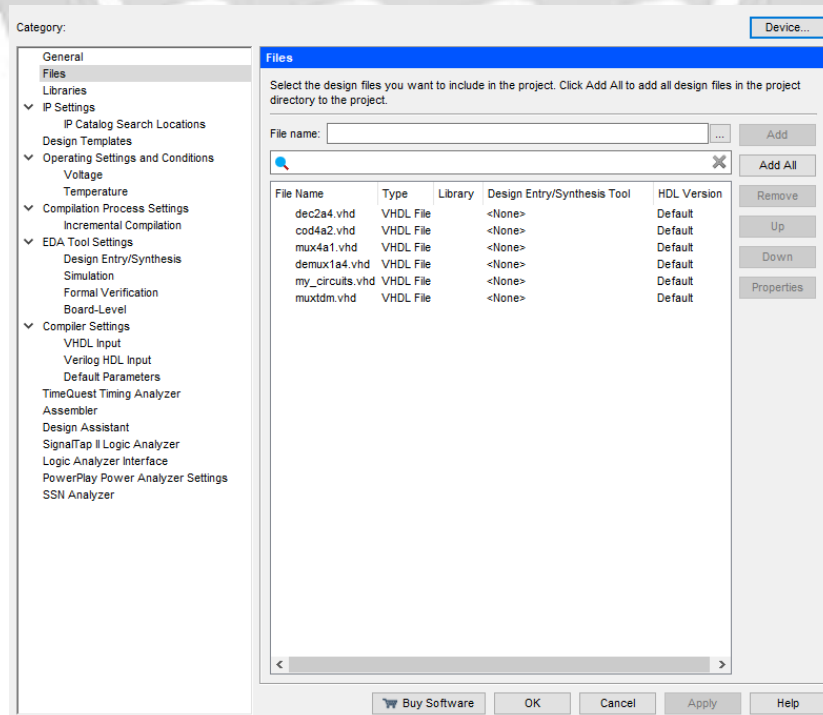
- iii. Cree un paquete de nombre `my_circuits.vhd`, en donde añada los componentes: `dec2a4`, `cod4a2`, `mux4a1` y `demux1a4`. Ese paquete debe ser añadido al proyecto.



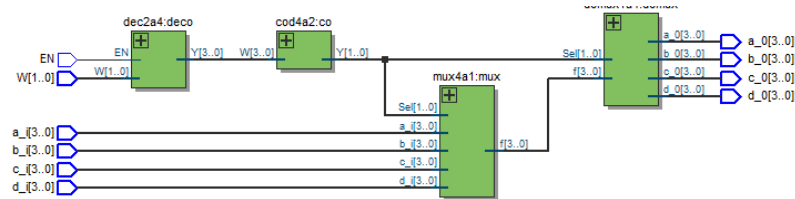
- iv. Realizar la descripción del circuito `muxtdm`, (archivo `muxtdm.vhd`). cuyas entradas son `a_i`, `b_i`, `c_i`, `d_i` de 4 bits cada una, `w` de 2 bits y `en` de 1 bit. Las salidas son `a_o`, `b_o`, `c_o` y `d_o` de 4 bits cada una.



- v. Aclaración: Antes de compilar (realizar la síntesis) entre en: Project -> add/remove files in Project. Asegúrese de que el orden de los archivos sea: dec2a4.vhd (parte superior), cod4a2.vhd, mux4a1.vhd, demux1a4.vhd, my_circuits.vhd y muxtdm.vhd (parte inferior), (cualquier cosa emplee los botones de Up o Down).



- vi. Interpretar el diseño del circuito muxtdm por medio de RTL Viewer.



- vii. Realizar la simulación del circuito muxtdm (añada las señales internas) y luego automaticase por medio del start.do (tal como indica la guía de la sesión N°03).

