DEEC
DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES
TÉCNICO LISBOA

# *Arquitectura de Computadores*

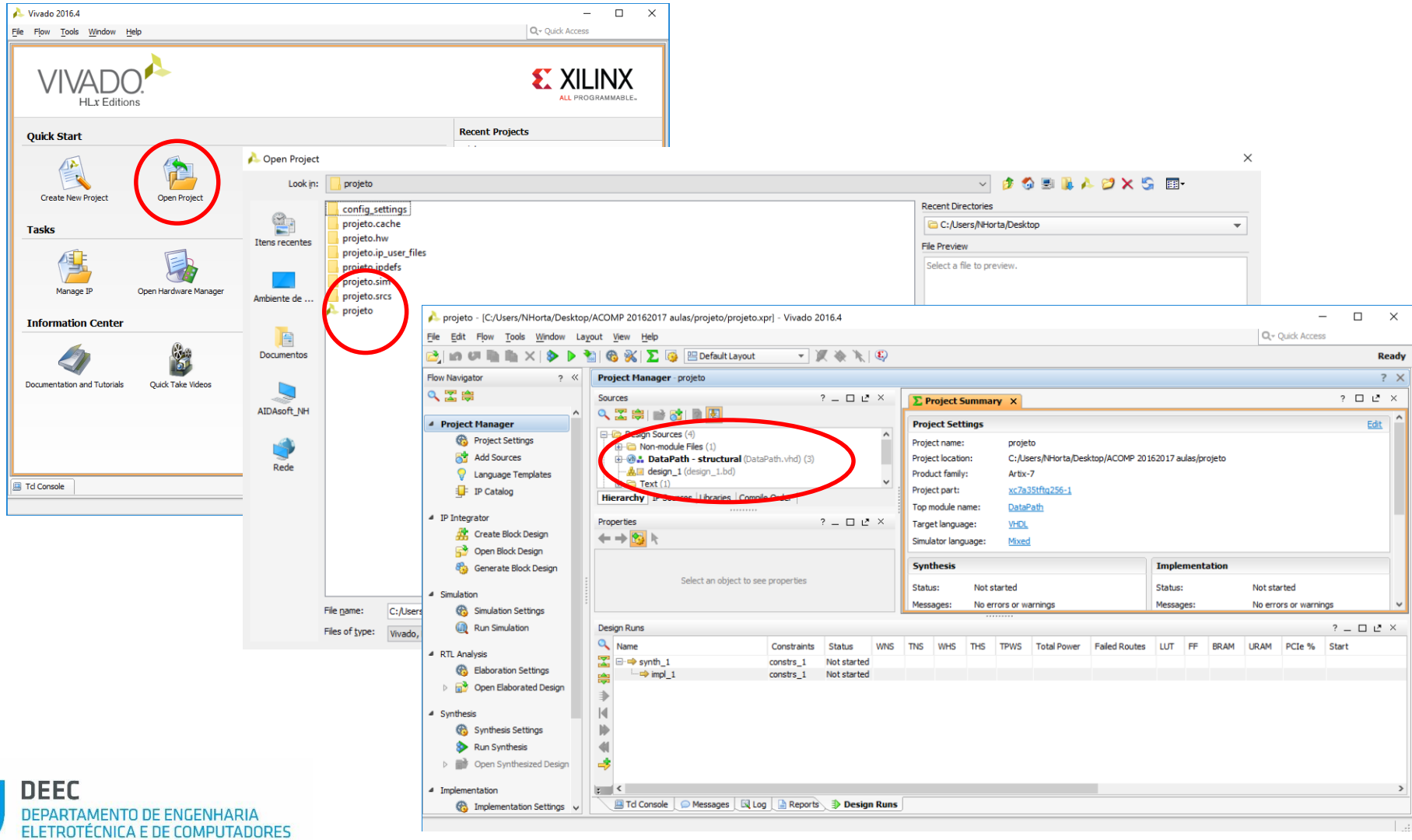
## *MEEC (2016/17 – 2º Sem.)*

## *Utilização do VIVADO*

**Prof. Nuno Horta**

# UNIDADE DE PROCESSAMENTO

## *Projeto no VIVADO (Abrir Projecto)*



# VHDL – Lab1

***DATAPATH***

- ***REGISTER***
- ***FUNCTIONALUNIT***
- ***DATAMEMORY***

DataPath.vhd

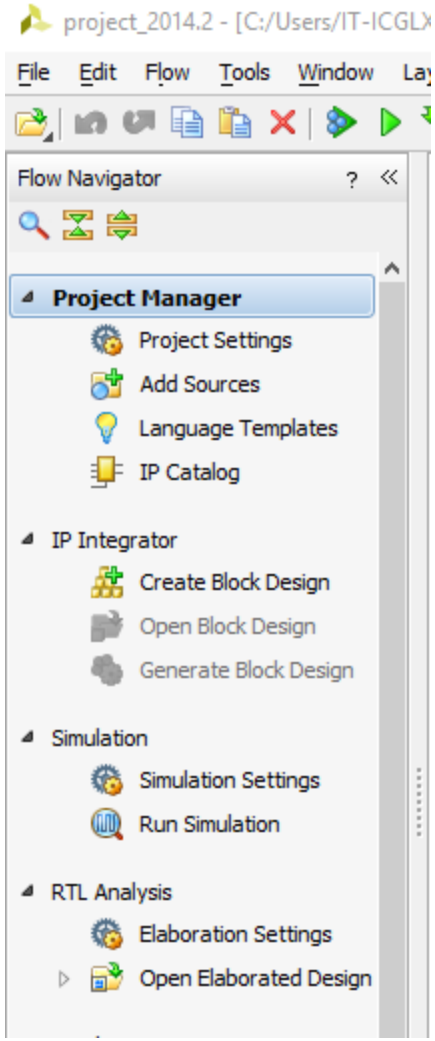
C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/DataPath.vhd

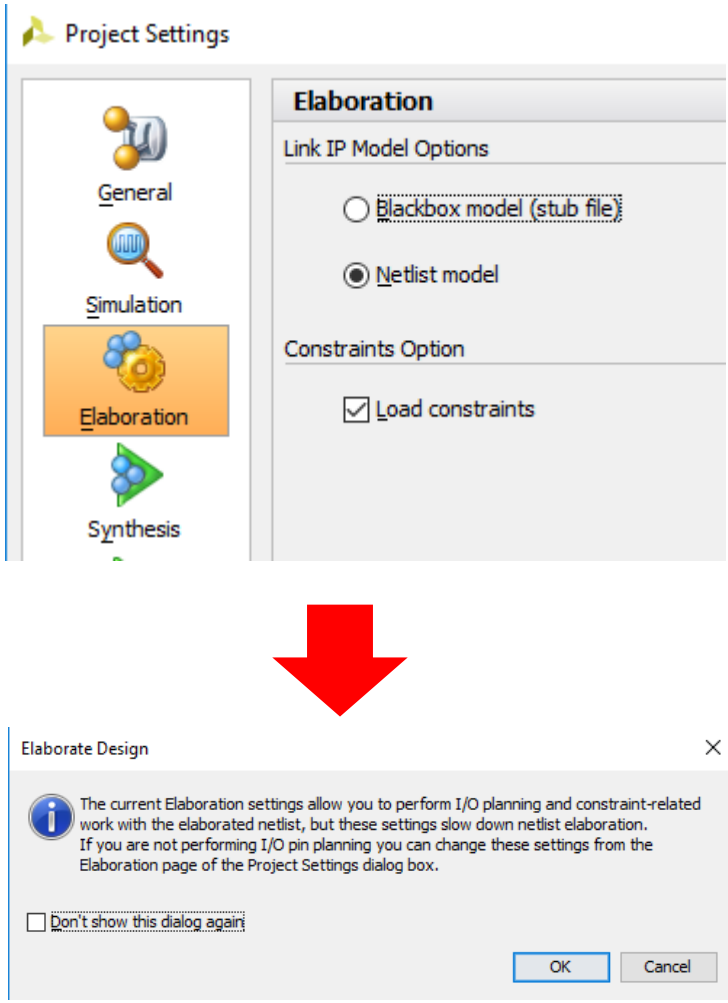
```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity DataPath is
5      Port (
6         AA : in std_logic_vector (3 downto 0);
7         BA : in std_logic_vector (3 downto 0);
8         DA : in std_logic_vector (3 downto 0);
9         WR : in std_logic;
10        MA : in std_logic;
11        MB : in std_logic;
12        KNS : in std_logic_vector (15 downto 0);
13        CLK : in std_logic;
14        MW : in std_logic;
15        FS : in std_logic_vector (3 downto 0);
16        MD : in std_logic;
17        D : out std_logic_vector (15 downto 0);
18        FL : out std_logic_vector (3 downto 0));
19 end DataPath;
20
21 architecture structural of DataPath is
22 component RegisterFile is
23     Port ( CLK : in std_logic;
24            Data : in std_logic_vector (15 downto 0);
25            WR : in std_logic;
26            DA : in std_logic_vector (3 downto 0);
27            AA : in std_logic_vector (3 downto 0);
28            BA : in std_logic_vector (3 downto 0);
29            A : out std_logic_vector (15 downto 0);
30            B : out std_logic_vector (15 downto 0));
31 end component;
32
33 component FunctionalUnit is
34     Port ( A : in std_logic_vector (15 downto 0);
35            B : in std_logic_vector (15 downto 0);
36            FS : in std_logic_vector (3 downto 0);
37            D : out std_logic_vector (15 downto 0);
38            FL : out std_logic_vector (3 downto 0));
39 end component;
40
41 component DataMemory is
42     Port ( Address : in std_logic_vector (15 downto 0);
43            Data : in std_logic_vector (15 downto 0);
44            WR : in std_logic;
45            CLK : in std_logic;
46            DataOut : out std_logic_vector (15 downto 0));
47 end component;
48
49 signal A, B, AK, BK, DM, DF, DI: std_logic_vector (15 downto 0);
50
51 begin
52     RF1 : RegisterFile port map (CLK => CLK, DATA=> DI, WR=>WR, DA=>DA, AA=>AA, BA=>BA, A=>A, B=>B);
53     AK <= A when MA='0' else
54           KNS when MA='1' else (others => 'X');
55     BK <= B when MB='0' else
56           KNS when MB='1' else (others => 'X');
57     FU1 : FunctionalUnit port map (A=>AK, B=>BK, FS=>FS, D=>DF, FL=>FL);
58     DM1 : DataMemory port map (Address=>BK, Data=>AK, WR=>MW, CLK=>CLK, DataOut=>DM);
59     DI <= DF when MD='0' else
60           DM when MD='1' else (others => 'X');
61     D <= DI;
62 end structural;
```

# VHDL – Lab1

***DATAPATH***

- ***REGISTER***
- ***FUNCTIONALUNIT***
- ***DATAMEMORY***



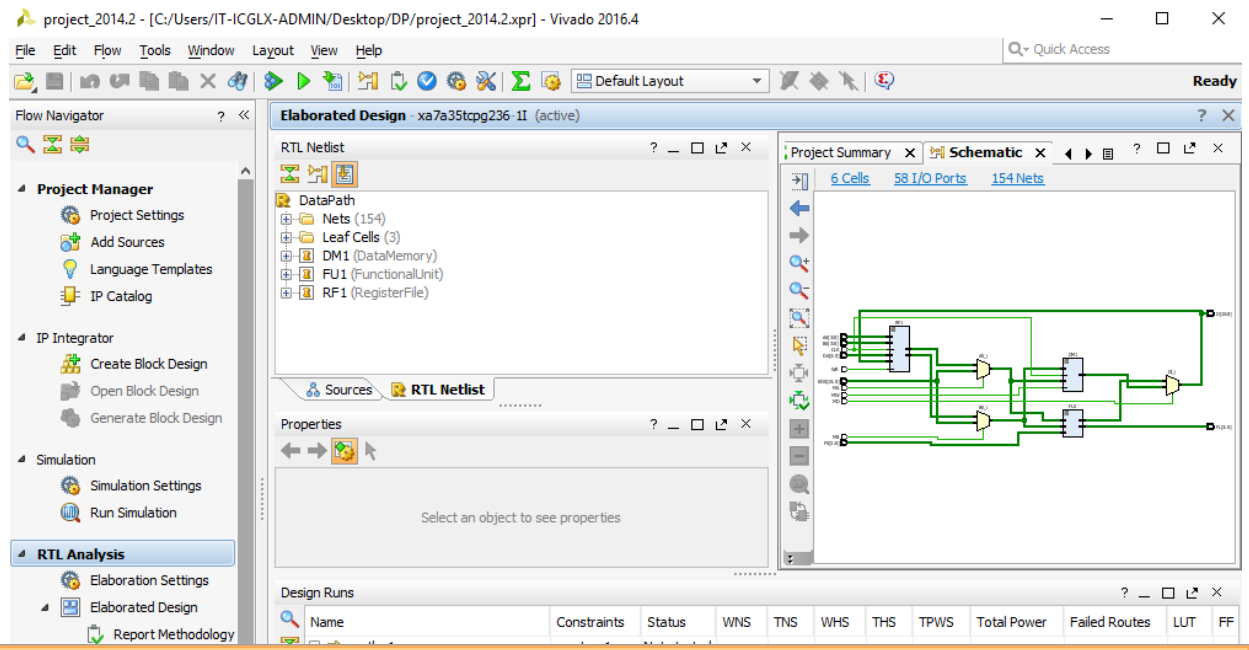
# VHDL – Lab1

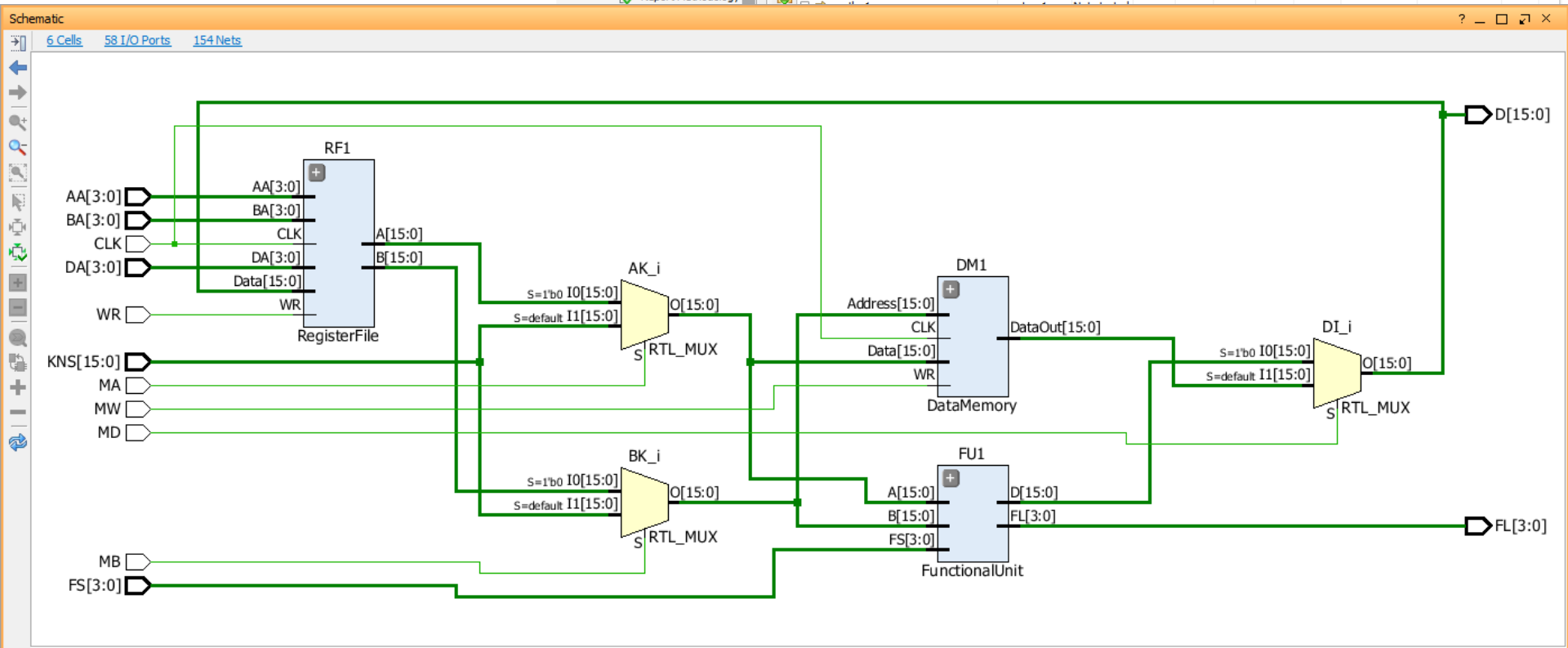
***DATAPATH***

***REGISTER***

***FUNCTIONALUNIT***

***DATAMEMORY***



# VHDL – L

***DATAPATH***

***REGISTERFILE***

***REGISTER16***

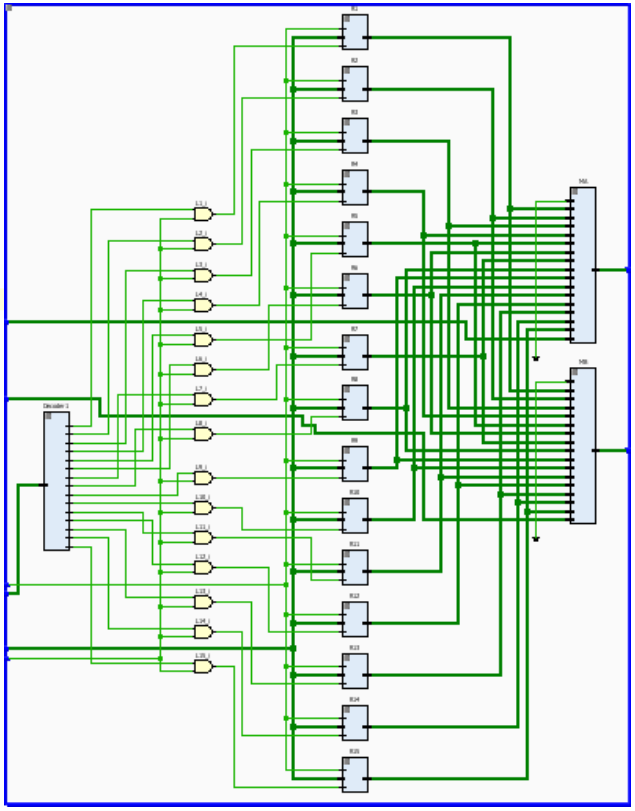
***DECODER***

***MUX16TO1***

RegisterFile.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/RegisterFile.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity RegisterFile is
5      Port ( CLK : in std_logic;
6             Data : in std_logic_vector (15 downto 0);
7             WR : in std_logic;
8             DA : in std_logic_vector (3 downto 0);
9             AA : in std_logic_vector (3 downto 0);
10            BA : in std_logic_vector (3 downto 0);
11            A : out std_logic_vector (15 downto 0);
12            B : out std_logic_vector (15 downto 0));
13 end RegisterFile;
14
15 architecture structural of RegisterFile is
16 component Register16 is
17     port(   D: in std_logic_vector(15 downto 0);
18             Load: in std_logic;
19             CLK: in std_logic;
20             Q: out std_logic_vector(15 downto 0):=(others => '0')
21             );
22 end component;
23
24 component mux16to1_16 is
25     Port ( S : in std_logic_vector (3 downto 0);
26            I0 : in std_logic_vector (15 downto 0);
27            I1 : in std_logic_vector (15 downto 0);
28            I2 : in std_logic_vector (15 downto 0);
29            I3 : in std_logic_vector (15 downto 0);
30            I4 : in std_logic_vector (15 downto 0);
31            I5 : in std_logic_vector (15 downto 0);
32            I6 : in std_logic_vector (15 downto 0);
33            I7 : in std_logic_vector (15 downto 0);
34            I8 : in std_logic_vector (15 downto 0);
35            I9 : in std_logic_vector (15 downto 0);
36            I10 : in std_logic_vector (15 downto 0);
37            I11 : in std_logic_vector (15 downto 0);
38            I12 : in std_logic_vector (15 downto 0);
39            I13 : in std_logic_vector (15 downto 0);
40            I14 : in std_logic_vector (15 downto 0);
41            I15 : in std_logic_vector (15 downto 0);
42            O : out std_logic_vector (15 downto 0));
43 end component;
44
45 component Decoder is
46     Port ( A : in std_logic_vector (3 downto 0);
47            D0 : out std_logic;
48            D1 : out std_logic;
49            D2 : out std_logic;
50            D3 : out std_logic;
51            D4 : out std_logic;
52            D5 : out std_logic;
53            D6 : out std_logic;
54            D7 : out std_logic;
55            D8 : out std_logic;
56            D9 : out std_logic;
57            D10 : out std_logic;
58            D11 : out std_logic;
59            D12 : out std_logic;
60            D13 : out std_logic;
61            D14 : out std_logic;
62            D15 : out std_logic);
63 end component;
64
```



```
65 signal D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15: std_logic;
66 signal L0, L1, L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13, L14, L15: std_logic;
67 signal Q0, Q1, Q2, Q3, Q4, Q5, Q6, Q7, Q8, Q9, Q10, Q11, Q12, Q13, Q14, Q15: std_logic_vector (15 downto 0);
68
69 begin
70     Decoder1: Decoder port map (A => DA, D0=>D0, D1=>D1, D2=>D2, D3=>D3, D4=>D4, D5=>D5, D6=>D6, D7=>D7, D8=>
71     L1 <= D1 and WR; L2 <= D2 and WR; L3 <= D3 and WR; L4 <= D4 and WR; L5 <= D5 and WR; L6 <= D6 and WR; L7
72     Q0 <= (others => '0');
73     R1: Register16 port map (D => Data, Load => L1, CLK => CLK, Q=> Q1);
74     R2: Register16 port map (D => Data, Load => L2, CLK => CLK, Q=> Q2);
75     R3: Register16 port map (D => Data, Load => L3, CLK => CLK, Q=> Q3);
76     R4: Register16 port map (D => Data, Load => L4, CLK => CLK, Q=> Q4);
77     R5: Register16 port map (D => Data, Load => L5, CLK => CLK, Q=> Q5);
78     R6: Register16 port map (D => Data, Load => L6, CLK => CLK, Q=> Q6);
79     R7: Register16 port map (D => Data, Load => L7, CLK => CLK, Q=> Q7);
80     R8: Register16 port map (D => Data, Load => L8, CLK => CLK, Q=> Q8);
81     R9: Register16 port map (D => Data, Load => L9, CLK => CLK, Q=> Q9);
82     R10: Register16 port map (D => Data, Load => L10, CLK => CLK, Q=> Q10);
83     R11: Register16 port map (D => Data, Load => L11, CLK => CLK, Q=> Q11);
84     R12: Register16 port map (D => Data, Load => L12, CLK => CLK, Q=> Q12);
85     R13: Register16 port map (D => Data, Load => L13, CLK => CLK, Q=> Q13);
86     R14: Register16 port map (D => Data, Load => L14, CLK => CLK, Q=> Q14);
87     R15: Register16 port map (D => Data, Load => L15, CLK => CLK, Q=> Q15);
88     MA: mux16to1_16 port map (S => AA, I0 => Q0, I1 => Q1, I2 => Q2, I3 => Q3, I4 => Q4, I5 => Q5, I6 => Q6,
89     MB: mux16to1_16 port map (S => BA, I0 => Q0, I1 => Q1, I2 => Q2, I3 => Q3, I4 => Q4, I5 => Q5, I6 => Q6,
90 end structural;
```

# VHDL – Lab1

***DATAPATH***

***REGISTERFILE***

***REGISTER16***

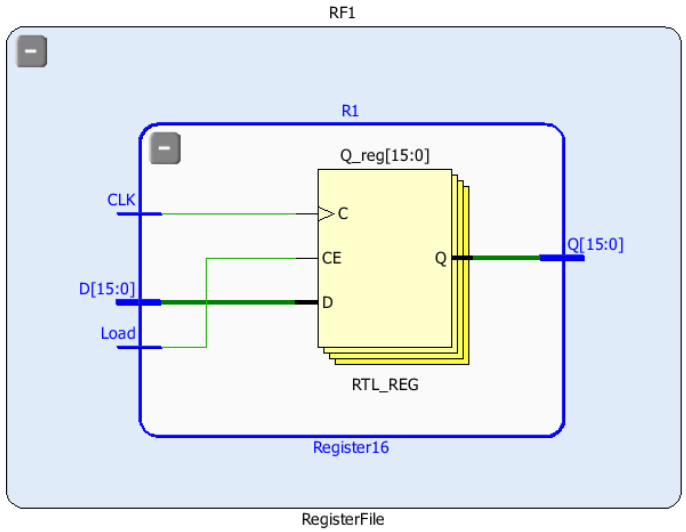
***DECODER***

***MUX16TO1***

Register16.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/imports/files/Register16.vhd

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  entity Register16 is
5      port(   D: in std_logic_vector(15 downto 0);
6              Load: in std_logic;
7              CLK: in std_logic;
8              Q: out std_logic_vector(15 downto 0):=(others => '0')
9              );
10 end Register16;
11
12 architecture structural of Register16 is
13 begin
14     Q <= D when CLK'event and CLK='1' and Load='1';
15 end structural;
16
```



# VHDL – Lab1

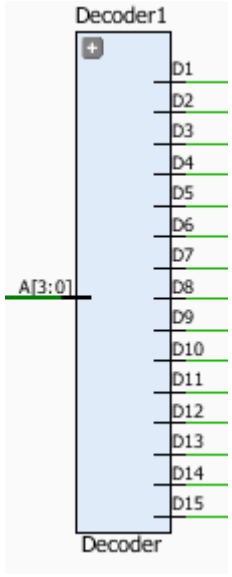
***DATAPATH***

***REGISTERFILE***

***REGISTER16***

***DECODER***

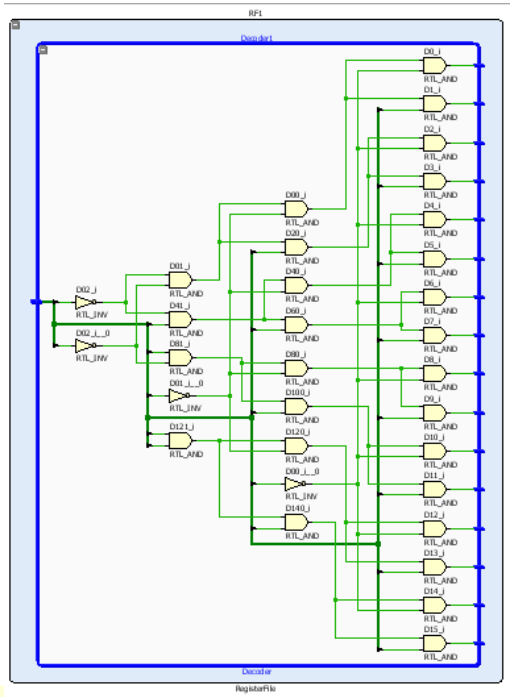
***MUX16TO1***




Decoder.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/Decoder.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity Decoder is
5      Port ( A : in STD_LOGIC_VECTOR (3 downto 0);
6             D0 : out STD_LOGIC;
7             D1 : out STD_LOGIC;
8             D2 : out STD_LOGIC;
9             D3 : out STD_LOGIC;
10            D4 : out STD_LOGIC;
11            D5 : out STD_LOGIC;
12            D6 : out STD_LOGIC;
13            D7 : out STD_LOGIC;
14            D8 : out STD_LOGIC;
15            D9 : out STD_LOGIC;
16            D10 : out STD_LOGIC;
17            D11 : out STD_LOGIC;
18            D12 : out STD_LOGIC;
19            D13 : out STD_LOGIC;
20            D14 : out STD_LOGIC;
21            D15 : out STD_LOGIC);
22 end Decoder;
23
24 architecture Behavioral of Decoder is
25 begin
26     D0  <= not A(3) and not A(2) and not A(1) and not A(0);
27     D1  <= not A(3) and not A(2) and not A(1) and     A(0);
28     D2  <= not A(3) and not A(2) and     A(1) and not A(0);
29     D3  <= not A(3) and not A(2) and     A(1) and     A(0);
30     D4  <= not A(3) and     A(2) and not A(1) and not A(0);
31     D5  <= not A(3) and     A(2) and not A(1) and     A(0);
32     D6  <= not A(3) and     A(2) and     A(1) and not A(0);
33     D7  <= not A(3) and     A(2) and     A(1) and     A(0);
34     D8  <=     A(3) and not A(2) and not A(1) and not A(0);
35     D9  <=     A(3) and not A(2) and not A(1) and     A(0);
36     D10 <=     A(3) and not A(2) and     A(1) and not A(0);
37     D11 <=     A(3) and not A(2) and     A(1) and     A(0);
38     D12 <=     A(3) and     A(2) and not A(1) and not A(0);
39     D13 <=     A(3) and     A(2) and not A(1) and     A(0);
40     D14 <=     A(3) and     A(2) and     A(1) and not A(0);
41     D15 <=     A(3) and     A(2) and     A(1) and     A(0);
42 end Behavioral;
43
```



DEEC
DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES
TÉCNICO LISBOA

# VHDL – Lab1

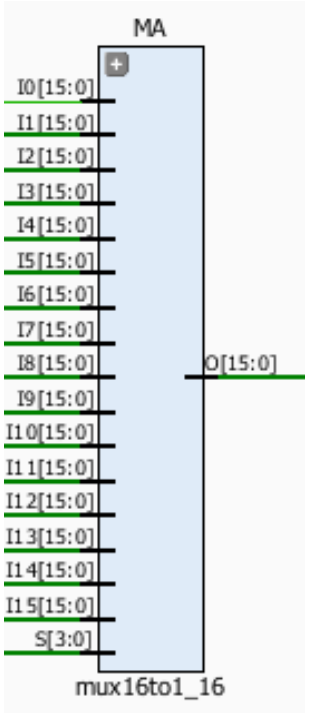
***DATAPATH***

***REGISTERFILE***

***REGISTER16***

***DECODER***

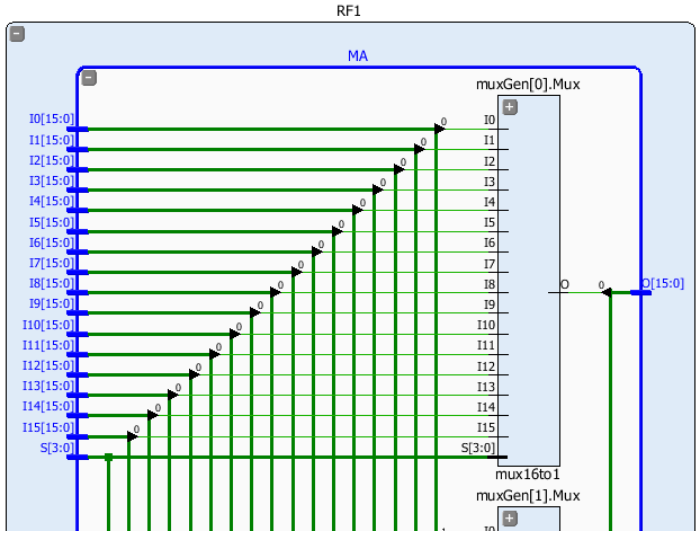
***MUX16TO1 (A)***




mux16to1_16.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/mux16to1_16.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity mux16to1_16 is
5      Port ( S : in std_logic_vector (3 downto 0);
6             I0 : in std_logic_vector (15 downto 0);
7             I1 : in std_logic_vector (15 downto 0);
8             I2 : in std_logic_vector (15 downto 0);
9             I3 : in std_logic_vector (15 downto 0);
10            I4 : in std_logic_vector (15 downto 0);
11            I5 : in std_logic_vector (15 downto 0);
12            I6 : in std_logic_vector (15 downto 0);
13            I7 : in std_logic_vector (15 downto 0);
14            I8 : in std_logic_vector (15 downto 0);
15            I9 : in std_logic_vector (15 downto 0);
16            I10 : in std_logic_vector (15 downto 0);
17            I11 : in std_logic_vector (15 downto 0);
18            I12 : in std_logic_vector (15 downto 0);
19            I13 : in std_logic_vector (15 downto 0);
20            I14 : in std_logic_vector (15 downto 0);
21            I15 : in std_logic_vector (15 downto 0);
22            O : out std_logic_vector (15 downto 0));
23 end mux16to1_16;
24
25 architecture structural of mux16to1_16 is
26 component mux16to1 is
27     Port ( S : in std_logic_vector (3 dow
28            I0 : in std_logic;
29            I1 : in std_logic;
30            I2 : in std_logic;
31            I3 : in std_logic;
32            I4 : in std_logic;
33            I5 : in std_logic;
34            I6 : in std_logic;
35            I7 : in std_logic;
36            I8 : in std_logic;
37            I9 : in std_logic;
38            I10 : in std_logic;
39            I11 : in std_logic;
40            I12 : in std_logic;
41            I13 : in std_logic;
42            I14 : in std_logic;
43            I15 : in std_logic;
44            O : out std_logic);
45 end component;
46
47 begin
48     muxGen: for I in 15 downto 0 generate
49         Mux: mux16to1 port map (S => S, I0 => I0(I), I1 => I1(I), I2 => I2(I), I3 =
50     end generate muxGen;
51
52 end structural;
```



# VHDL – Lab1

***DATAPATH***

***REGISTERFILE***

***REGISTER16***

***DECODER***

***MUX16TO1 (B)***

mux16to1.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/mux16to1.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity mux16to1 is
5      Port ( S : in std_logic_vector (3 downto 0);
6             I0 : in std_logic;
7             I1 : in std_logic;
8             I2 : in std_logic;
9             I3 : in std_logic;
10            I4 : in std_logic;
11            I5 : in std_logic;
12            I6 : in std_logic;
13            I7 : in std_logic;
14            I8 : in std_logic;
15            I9 : in std_logic;
16            I10 : in std_logic;
17            I11 : in std_logic;
18            I12 : in std_logic;
19            I13 : in std_logic;
20            I14 : in std_logic;
21            I15 : in std_logic;
22            O : out std_logic);
23 end mux16to1;
24
25 architecture structural of mux16to1 is
26 begin
27 O <=     (I0 and not S(3) and not S(2) and not S(1) and not S(0)) or
28         (I1 and not S(3) and not S(2) and not S(1) and     S(0)) or
29         (I2 and not S(3) and not S(2) and     S(1) and not S(0)) or
30         (I3 and not S(3) and not S(2) and     S(1) and     S(0)) or
31         (I4 and not S(3) and     S(2) and not S(1) and not S(0)) or
32         (I5 and not S(3) and     S(2) and not S(1) and     S(0)) or
33         (I6 and not S(3) and     S(2) and     S(1) and not S(0)) or
34         (I7 and not S(3) and     S(2) and     S(1) and     S(0)) or
35         (I8 and     S(3) and not S(2) and not S(1) and not S(0)) or
36         (I9 and     S(3) and not S(2) and not S(1) and     S(0)) or
37         (I10 and    S(3) and not S(2) and     S(1) and not S(0)) or
38         (I11 and    S(3) and not S(2) and     S(1) and     S(0)) or
39         (I12 and    S(3) and     S(2) and not S(1) and not S(0)) or
40         (I13 and    S(3) and     S(2) and not S(1) and     S(0)) or
41         (I14 and    S(3) and     S(2) and     S(1) and not S(0)) or
42         (I15 and    S(3) and     S(2) and     S(1) and     S(0));
43
44 end structural;
```

# VHDL

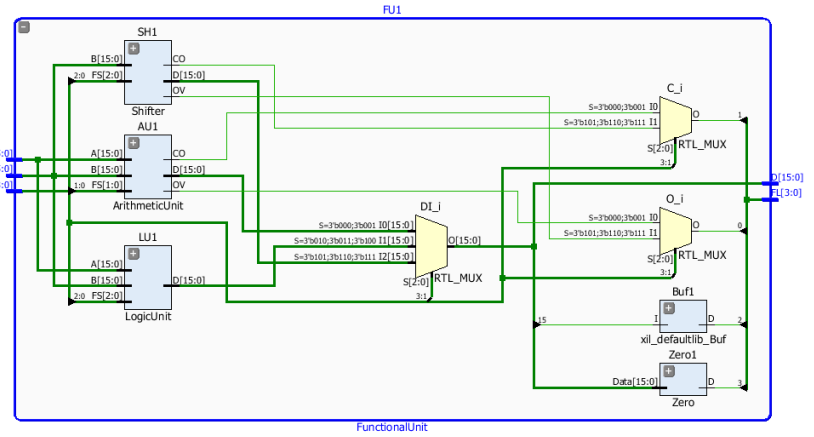
## *DATAPATH*

## *FUNCTION*

FunctionalUnit.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/FunctionalUnit.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity FunctionalUnit is
5      Port ( A : in std_logic_vector (15 downto 0);
6             B : in std_logic_vector (15 downto 0);
7             FS : in std_logic_vector (3 downto 0);
8             D : out std_logic_vector (15 downto 0);
9             FL : out std_logic_vector (3 downto 0));
10 end FunctionalUnit;
11
12 architecture structural of FunctionalUnit is
13 component ArithmeticUnit is
14     Port ( A : in std_logic_vector (15 downto 0);
15            B : in std_logic_vector (15 downto 0);
16            FS : in std_logic_vector (1 downto 0);
17            D : out std_logic_vector (15 downto 0);
18            CO : out std_logic;
19            OV : out std_logic);
20 end component;
21
22 component LogicUnit is
23     Port ( A : in std_logic_vector (15 downto 0);
24            B : in std_logic_vector (15 downto 0);
25            FS : in std_logic_vector (2 downto 0);
26            D : out std_logic_vector (15 downto 0)
27 end component;
28
29 component Shifter is
30     Port ( B : in std_logic_vector (15 downto 0);
31            FS : in std_logic_vector (2 downto 0);
32            D : out std_logic_vector (15 downto 0)
33            CO : out std_logic;
34            OV : out std_logic);
35 end component;
36
37 component Zero is
38     port(   Data : in std_logic_vector (15 downto
39 end component;
40
41 component Buf is
42     port(   I : in std_logic;  D : out std_logic);
43 end component;
44
45 signal DI : std_logic_vector (15 downto 0);
46 signal DA : std_logic_vector (15 downto 0);
47 signal DL : std_logic_vector (15 downto 0);
48 signal DS : std_logic_vector (15 downto 0);
49 signal COA, OVA, COS, OVS : std_logic;
50 signal Z, N, C, O : std_logic;
```




```
51
52 begin
53     AU1 : ArithmeticUnit port map (A => A, B=>B, FS=> FS(1 downto 0), D=>DA, CO=>CO
54     LU1 : LogicUnit port map (A => A, B=>B, FS=> FS(2 downto 0), D=>DL);
55     SH1 : Shifter port map (B=>B, FS=>FS(2 downto 0), D=>DS, CO=>COS, OV=>OVS);
56
57     with FS(3 downto 1) select
58     DI <=   DA when "000" | "001",
59             DL when "010" | "011" | "100",
60             DS when "101" | "110" | "111",
61             (others => 'X') when others;
62
63     -- zero
64     Zero1: Zero port map (Data=>DI, D=>Z);
65
66     -- negative
67     Buf1: Buf port map (I=> DI(15), D=> N);
68
69     -- cary
70     with FS(3 downto 1) select
71     C <=    COA when "000" | "001",
72                 '-' when "010" | "011" | "100",
73                 COS when "101" | "110" | "111",
74                 'X' when others;
75
76     -- overflow
77     with FS(3 downto 1) select
78     O <=    OVA when "000" | "001",
79                 '-' when "010" | "011" | "100",
80                 OVS when "101" | "110" | "111",
81                 'X' when others;
82
83      D <= DI;
84      FL <= Z & N & C & O;
85 end structural;
```


DEEC
DEPARTAMENTO DE ENG
ELETROTÉCNICA E DE CO
TÉCNICO LISBOA

# VHDL – Lab

***DATAPATH***

***FUNCTIONALUNIT***

***ARITMETICUNIT***

ArithmeticUnit.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/ArithmeticUnit.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity ArithmeticUnit is
5      Port ( A : in std_logic_vector (15 downto 0);
6             B : in std_logic_vector (15 downto 0);
7             FS : in std_logic_vector (1 downto 0);
8             D : out std_logic_vector (15 downto 0);
9             CO : out std_logic;
10            OV : out std_logic);
11 end ArithmeticUnit;
12
13 architecture structural of ArithmeticUnit is
14 component ArithmeticUnitI is
15     Port ( Ai : in std_logic;
16        Bi : in std_logic;
17        FS : in std_logic_vector(1 downto 0);
18        CI : in std_logic;
19        Di : out std_logic;
20        CO : out std_logic);
21 end component;
22 signal Yn : std_logic;
23 signal Z : std_logic_vector (15 downto 0);
24 signal C : std_logic_vector (16 downto 0);
25
26 begin
27     C(0) <= FS(0);
28     aritGen: for I in 15 downto 0 generate
29         AUi: ArithmeticUnitI port map (Ai=>A(I), Bi=>B(I), FS=>FS, CI=>C(I), Di=>Z(
30     end generate aritGen;
31     D <= Z;
32
33     -- cary
34     CO <= C(16);
35     -- overflow
36     Yn <= (B(15) and not FS(1)) or (not B(15) and FS(1));
37     OV <= (A(15) and Yn and not Z(15)) or (not A(15) and not Yn and Z(15));
38
39 end structural;
```

# VHDL – Lab1

***DATAPATH***

***FUNCTIONALUNIT***

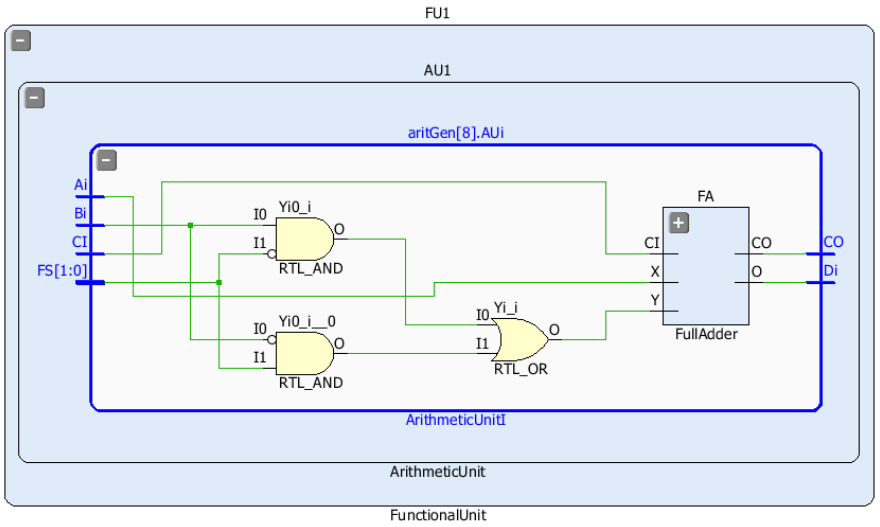
***ARITMETICUNIT***

***ARUint***

ArithmeticUnitI.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/ArithmeticUnitI.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity ArithmeticUnitI is
5      Port ( Ai : in std_logic;
6         Bi : in std_logic;
7         FS : in std_logic_vector(1 downto 0);
8         CI : in std_logic;
9         Di : out std_logic;
10        CO : out std_logic);
11 end ArithmeticUnitI;
12
13 architecture structural of ArithmeticUnitI is
14 component FullAdder is
15     port(X: in std_logic;
16          Y: in std_logic;
17          CI: in std_logic;
18          O: out std_logic;
19          CO: out std_logic);
20 end component;
21 signal Yi : std_logic;
22 begin
23     Yi <= (Bi and not FS(1)) or (not Bi and FS(1));
24     FA: FullAdder port map (X => Ai, Y => Yi, CI => CI, O => Di, CO => CO);
25 end structural;
26
```



# VHDL – Lab1

***DATAPATH***

***FUNCTIONALUNIT***

***ARITMETICUNIT***

***ARUint***

***FULLADDER***

FullAdder.vhd

...p/DP/project_2014.2.srcs/sources_1/imports/sources_1/imports/nibbles/Full...

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  entity FullAdder is
5      port(   X: in std_logic;
6          Y: in std_logic;
7          CI: in std_logic;
8          O: out std_logic;
9          CO: out std_logic);
10 end FullAdder;
11
12 architecture structural of FullAdder is
13     signal N00:  std_logic;
14 begin
15     N00 <= X xor Y;
16     O   <= N00 xor CI;
17     CO  <= (X and Y) or (N00 and CI);
18 end structural;
```

# VHDL – Lab1

***DATAPATH***

***FUNCTIONALUNIT***

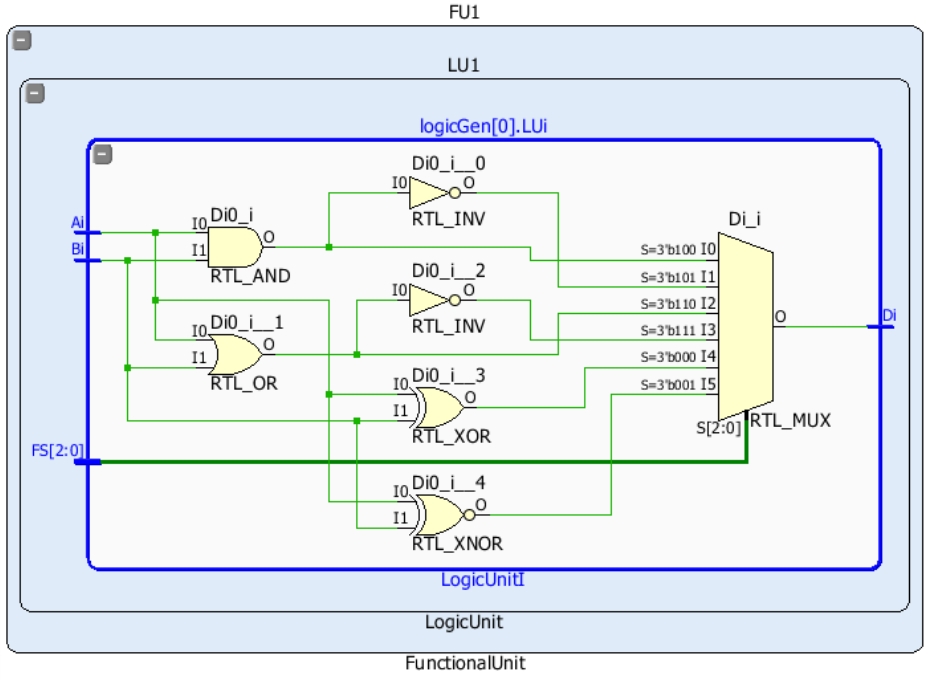
***LOGICUNIT***

***LUint***

LogicUnit.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/LogicUnit.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity LogicUnit is
5      Port ( A : in std_logic_vector (15 downto 0);
6             B : in std_logic_vector (15 downto 0);
7             FS : in std_logic_vector (2 downto 0);
8             D : out std_logic_vector (15 downto 0));
9  end LogicUnit;
10
11 architecture structural of LogicUnit is
12 component LogicUnitI is
13     Port ( Ai : in std_logic;
14        Bi : in std_logic;
15        FS : in std_logic_vector (2 downto 0);
16        Di : out std_logic);
17 end component;
18 begin
19     logicGen: for I in 15 downto 0 generate
20         LUi : LogicUnitI port map (Ai=>A(I), Bi=>B(I), FS=>FS, Di=>D(I));
21     end generate logicGen;
22 end structural;
```




LogicUnitI.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/LogicUnitI.vhd

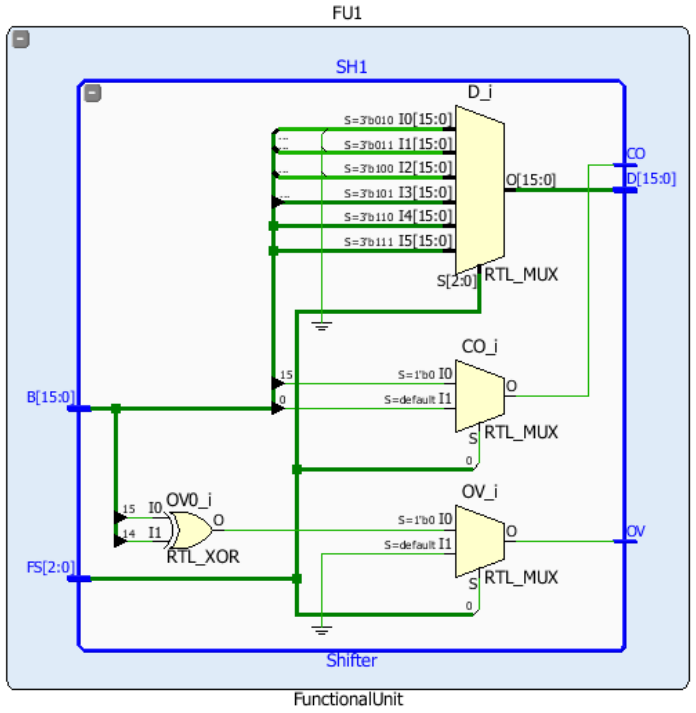
```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity LogicUnitI is
5      Port ( Ai : in std_logic;
6         Bi : in std_logic;
7         FS : in std_logic_vector (2 downto 0);
8         Di : out std_logic);
9  end LogicUnitI;
10
11 architecture structural of LogicUnitI is
12 begin
13     with FS select
14     Di <= Ai and Bi      when "100",
15         Ai nand Bi       when "101",
16         Ai or Bi         when "110",
17         Ai nor Bi        when "111",
18         Ai xor Bi        when "000",
19         Ai xnor Bi       when "001",
20         '-'              when "010" | "011",
21         'X'              when others;
22 end structural;
```

# VHDL – Lab1

***DATAPATH***

***FUNCTIONALUNIT***

***SHIFTER***




Shifter.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sourc

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity Shifter is
5      Port ( B : in std_logic_vector (15 downto 0);
6             FS : in std_logic_vector (2 downto 0);
7             D : out std_logic_vector (15 downto 0);
8             CO : out std_logic;
9             OV : out std_logic);
10 end Shifter;
11
12 architecture structural of Shifter is
13 begin
14     with FS select
15     D <= (others => '-')          when "000" | "001",
16          B(14 downto 0) & '0'     when "010",
17          '0' & B(15 downto 1)     when "011",
18          B(14 downto 0) & '0'     when "100",
19          B(15) & B(15 downto 1)   when "101",
20          B(14 downto 0) & B(15)   when "110",
21          B(0) & B(15 downto 1)    when "111",
22          (others => 'X')          when others;
23
24      CO <= B(15) when FS(0)='0' else
25            B(0) when FS(0)='1' else
26            'X';
27
28      OV <=  B(15) xor B(14) when FS(0)='0' else
29             '0' when FS(0)='1' else
30             'X';
31
32 end structural;
```

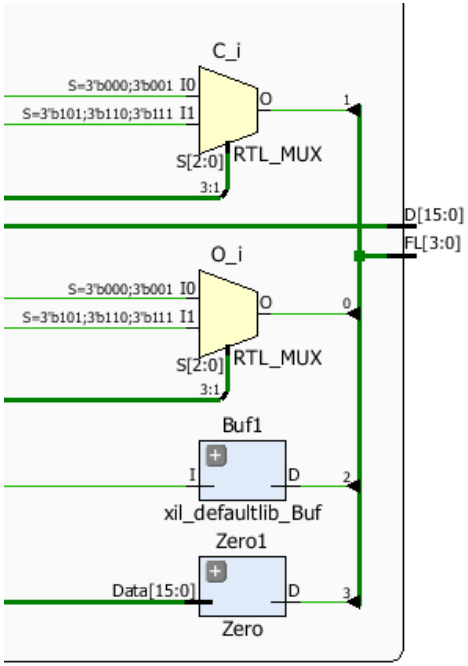
# VHDL – Lab1

***DATAPATH***

***FUNCTIONALUNIT***

***Zero1***

***Buf1***




Zero.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/Zero.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity Zero is
5      port(   Data : in std_logic_vector (15 downto 0);
6              D : out std_logic);
7  end Zero;
8
9  architecture Behavioral of Zero is
10 begin
11     D <= ((not Data(0) and not Data(1) and not Data(2) and not Data(3)) and
12         ((not Data(8) and not Data(9) and not Data(10) and not Data(11)) and
13 end Behavioral;
14
```

Buf.vhd

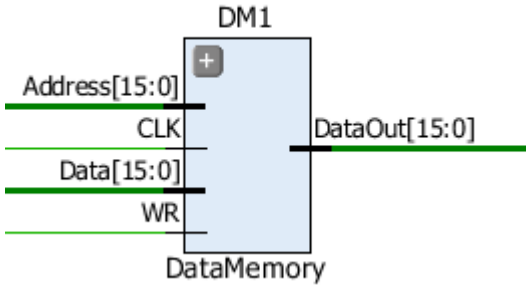
C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/im

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity Buf is
5      port(   I : in std_logic;
6              D : out std_logic);
7  end Buf;
8
9  architecture Behavioral of Buf is
10 begin
11     D <= I;
12 end Behavioral;
```

# VHDL – Lab1

***DATAPATH***

***DATAMEM***




DataMemory.vhd

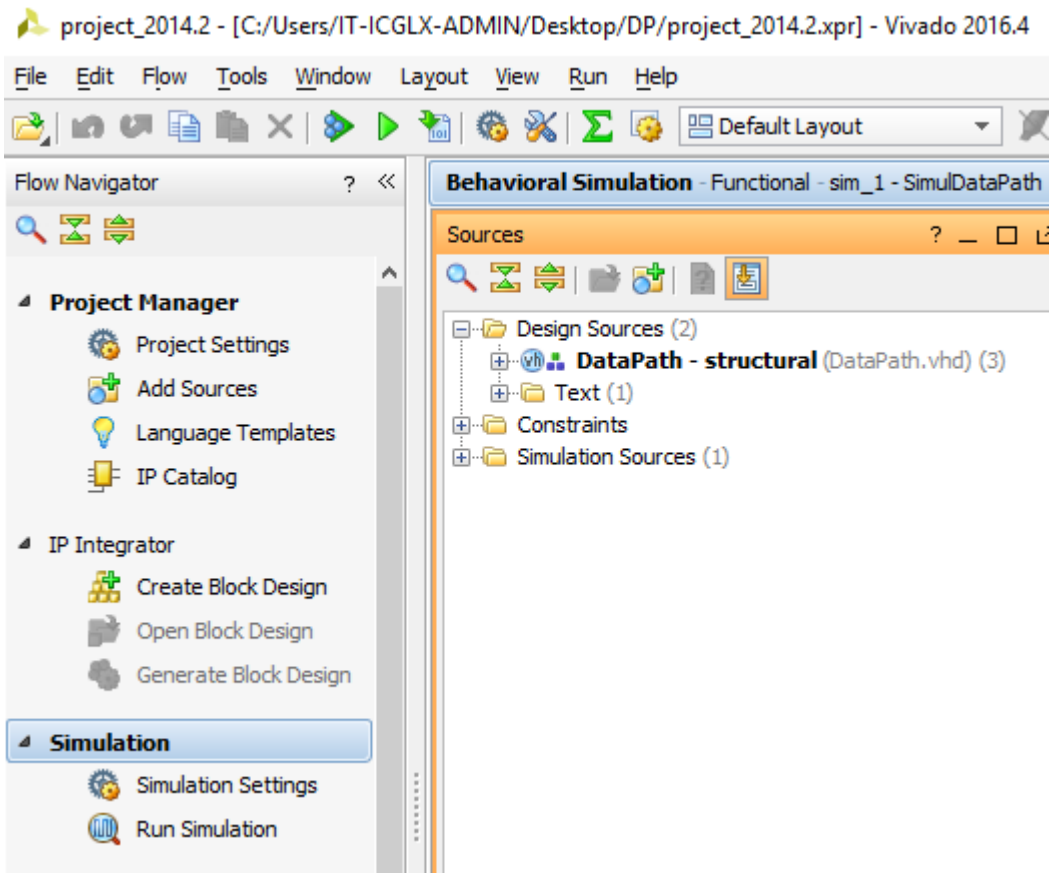
C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sources_1/imports/sources_1/new/DataMemory.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use ieee.numeric_std.all;
4
5  entity DataMemory is
6      Port ( Address : in std_logic_vector (15 downto 0);
7             Data : in std_logic_vector (15 downto 0);
8             WR : in std_logic;
9             CLK : in std_logic;
10            DataOut : out std_logic_vector (15 downto 0));
11 end DataMemory;
12
13 architecture structural of DataMemory is
14 type storage_type is array (0 to 255) of std_logic_vector(15 downto 0);
15 signal storage: storage_type := (
16      0 => x"1000",       --
17      1 => x"0001",       --
18      2 => x"0111",       --
19      3 => x"1010",       --
20      4 => x"1111",       --
21      5 => x"0100",       --
22      6 => x"1110",       --
23      7 => x"0000",       --
24      8 => x"1000",       --
25      others => x"0000");
26 begin
27     DataOut <= storage(to_integer(unsigned(Address(7 downto 0)))) when not Is_X(Address) else (others=>'X');
28     storage(to_integer(unsigned(Address(7 downto 0))))<=Data when CLK'event and CLK='1' and WR='1' and not Is_X(Address);
29 end structural;
```

Conteúdo da Memória

# VHDL – Lab1 (SIM)

***DATAPATH***



# VHDL – Lab1 (SIM)

***DATAPATH***

SimulDataPath.vhd

C:/Users/IT-ICGLX-ADMIN/Desktop/DP/project_2014.2.srcs/sim_1/imports/new/SimulDataPath.vhd

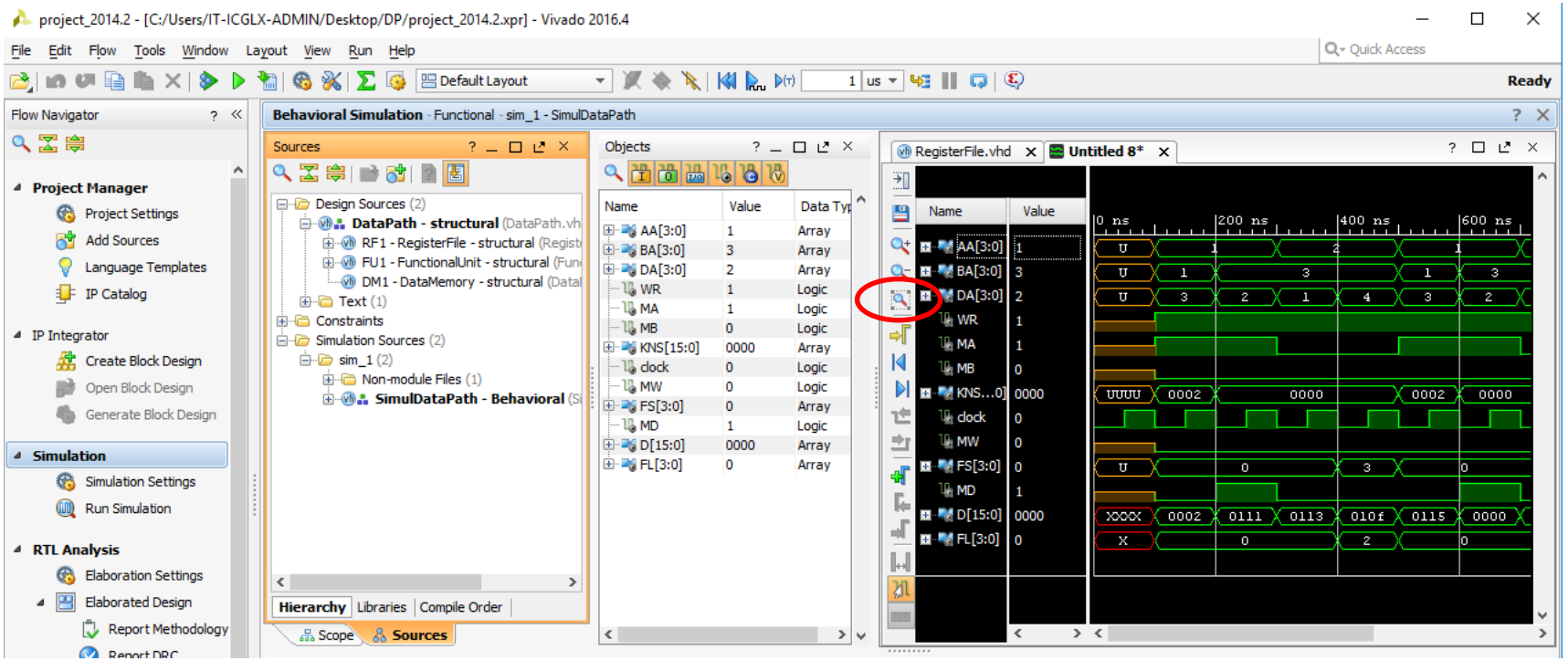
```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use ieee.numeric_std.all;
4
5  entity SimulDataPath is
6  end SimulDataPath;
7
8  architecture Behavioral of SimulDataPath is
9
10 function to_LVU(arg, size : natural) return std_logic_vector is
11 begin
12     return std_logic_vector(to_unsigned(arg, size));
13 end;
14
15 component DataPath is
16     Port (
17        AA : in std_logic_vector (3 downto 0);
18        BA : in std_logic_vector (3 downto 0);
19        DA : in std_logic_vector (3 downto 0);
20        WR : in std_logic;
21        MA, MB : in std_logic;
22        KNS : in std_logic_vector (15 downto 0)
23        CLK : in std_logic;
24        MW : in std_logic;
25        FS : in std_logic_vector (3 downto 0);
26        MD : in std_logic;
27        D : out std_logic_vector (15 downto 0);
28        FL : out std_logic_vector (3 downto 0))
29 end component;
30
31 signal  AA : std_logic_vector (3 downto 0);
32 signal  BA : std_logic_vector (3 downto 0);
33 signal  DA : std_logic_vector (3 downto 0);
34 signal  WR : std_logic;
35 signal  MA, MB : std_logic;
36 signal  KNS : std_logic_vector (15 downto 0);
37 signal  clock : std_logic := '0';
38 signal  MW : std_logic;
39 signal  FS : std_logic_vector (3 downto 0);
40 signal  MD : std_logic;
41 signal  D : std_logic_vector (15 downto 0);
42 signal  FL : std_logic_vector (3 downto 0);
43
```

```
44 begin
45     process
46     begin
47         wait for 50 ns;
48         clock <= not clock;
49     end process;
50
51     process
52     begin
53         wait until clock'event and clock='0';
54         -- R3 <- 2
55         AA<=x"1" ; BA<=x"1"; DA<=x"3"; WR<='1'; MA<='1'; MB<='0'; KNS<=x"0002"; MW<='0'; FS<=x"0"; MD<='0';
56         wait until clock'event and clock='0';
57         -- R2 <- M[2]
58         AA<=x"1" ; BA<=x"3"; DA<=x"2"; WR<='1'; MA<='1'; MB<='0'; KNS<=x"0000"; MW<='0'; FS<=x"0"; MD<='1';
59         wait until clock'event and clock='0';
60         -- R1 <- R2 + R3
61         AA<=x"2" ; BA<=x"3"; DA<=x"1"; WR<='1'; MA<='0'; MB<='0'; KNS<=x"0000"; MW<='0'; FS<=x"0"; MD<='0';
62         wait until clock'event and clock='0';
63         -- R4 <- R2 - R3
64         AA<=x"2" ; BA<=x"3"; DA<=x"4"; WR<='1'; MA<='0'; MB<='0'; KNS<=x"0000"; MW<='0'; FS<=x"3"; MD<='0';
65 end process;
66
67     DP0: DataPath port map (
68         AA=>AA, BA=>BA, DA=>DA, WR=>WR, MA=>MA, MB=>MB, KNS=>KNS, MW=>MW, FS=>FS, MD=>MD, D=>D, FL=>FL, CLK=>
69         );
70
71 end Behavioral;
```

R3 <- 2
R2 <- M[R3]
R1 <- R2+R3
R4 <- R2-R3

# VHDL – Lab1 (SIM)

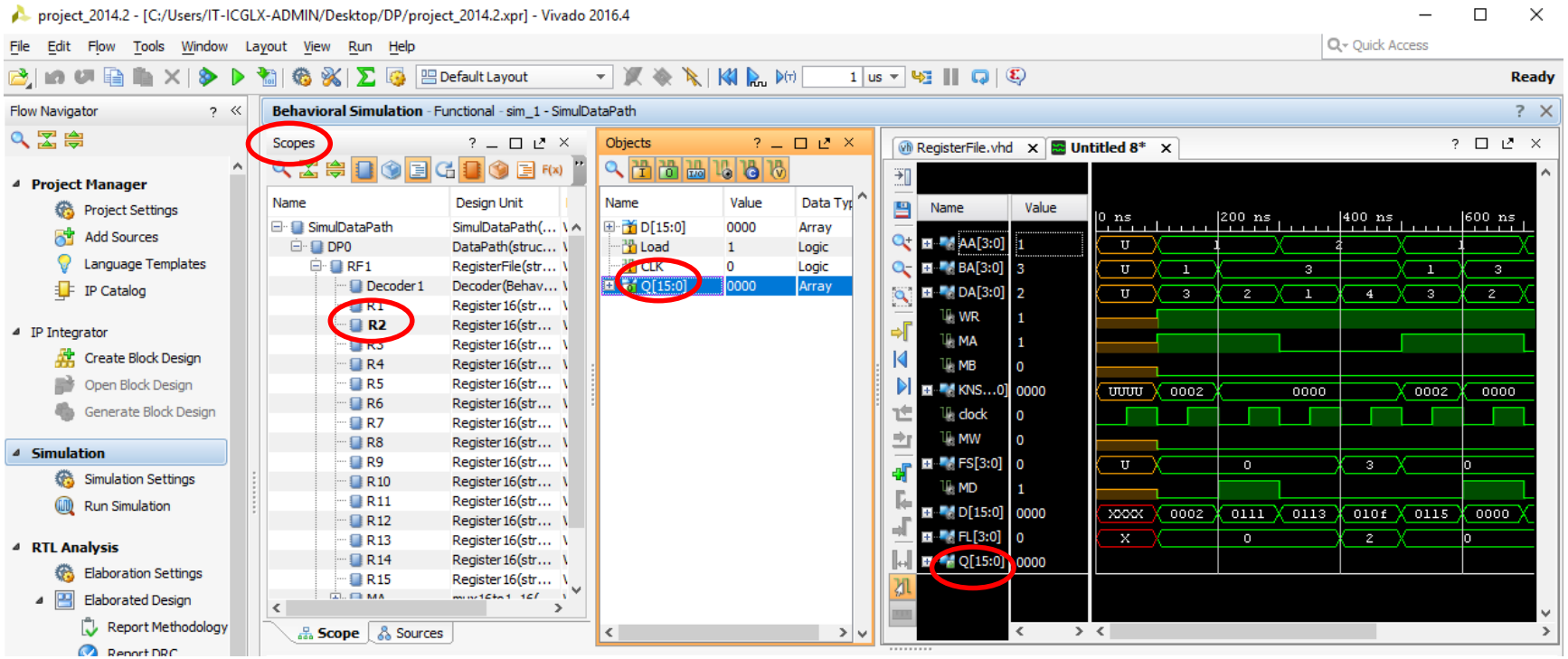
## *DATAPATH*



# VHDL – Lab1 (SIM)

## *DATAPATH*

### *Adicionar variável à simulação*



# VHDL – Lab1 (SIM)

## *DATAPATH*