Lab 2: Introduction to VHDL

Date: 03.10.2022

Name: Orkun İbrahim Kök

Section: EE102-01

Purpose

The purpose of this experiment is to get used to how to implement logic circuits into Vivado and combining these logic circuits with BASYS3.

Design specifications

There are 2 inputs and one output in the combinational logic circuit that I designed.

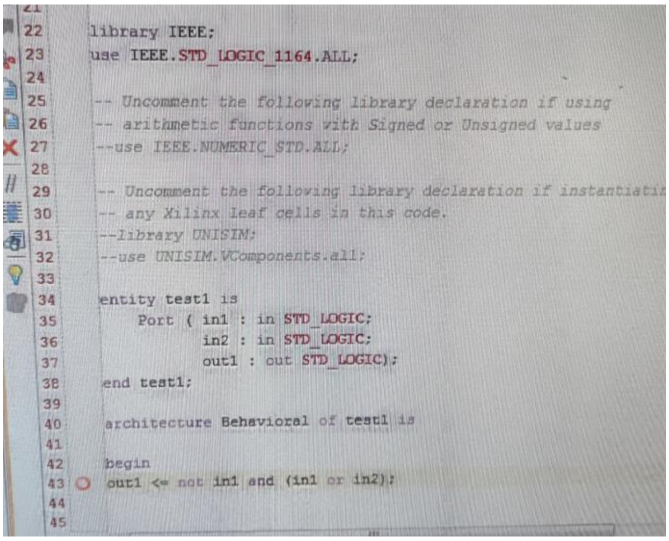
Input 1	Input 2	Output
A	B	A’(A+B)

Methodology

First I created a combinational logic circuit by my own before coming to lab. The inputs of the logic circuit are A and B, and the output is A’(A+B). I used 1 AND gate, 1 OR gate and, 1 NOT gate. After creating the circuit, I transferred it to Vivado. Entering the inputs and outputs, the scheme of the logic is appeared on the Vivado. After, according to the truth table of the combinational logic circuit, I made the time diagram. Then I move on to the second part of the experiment which is demonstrating the logic circuit on the BASYS3. I created a constraint file in order to assign the inputs as switches and outputs as LEDs on the BASYS3.

Q1) How does one specify the inputs and outputs of a module in VHDL?

In Vivado, while writing the VHDL code, inputs and outputs can be assigned as:



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

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiati
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity test1 is
    Port ( in1 : in STD_LOGIC;
           in2 : in STD_LOGIC;
           out1 : out STD_LOGIC);
end test1;

architecture Behavioral of test1 is

begin
out1 <= not in1 and (in1 or in2);
```

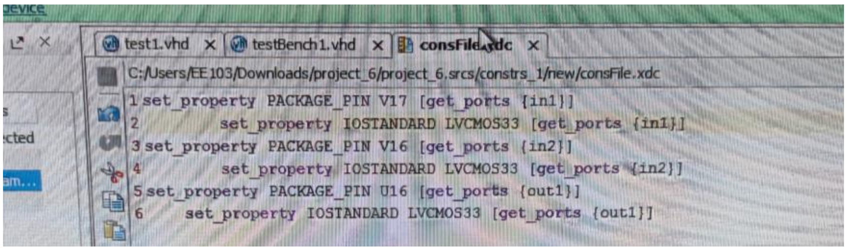
Q2) How does one use a module inside another code/module? What does PORT MAP do?

By indentation a module can be used in another module, without breaking the hierarchy. A port map connects the interface of a component to the reset of the design.

Q3) What is a constraint file? How does it relate your code to the pins on your FPGA?

Constraint file is a list of VHDL code in order to determine the input(s) and output(s) on the BASYS3. For example, the constraint file that I wrote takes input1 and input2 as V16 and V17 switches, and output as LED U16 (Figure: 1). Overall, constraint file makes the connection between the logic circuit and pins of the FPGA.

Figure: 1 Constraint file



C:/Users/EE103/Downloads/project_6/project_6.srcs/constrs_1/new/consFile.xdc

```
set_property PACKAGE_PIN V17 [get_ports {in1}]
        set_property IOSTANDARD LVCMOS33 [get_ports {in1}]
set_property PACKAGE_PIN V16 [get_ports {in2}]
        set_property IOSTANDARD LVCMOS33 [get_ports {in2}]
set_property PACKAGE_PIN U16 [get_ports {out1}]
    set_property IOSTANDARD LVCMOS33 [get_ports {out1}]
```

Q4) What is the purpose of writing a testbench?

After writing the VHDL code of the logic circuit, it has to be tested, therefore it is tested by writing a testbench code. In testbench, it is checked whether or not the design is working properly. Also, one other purpose can be viewing the waveform of the logical circuit.

Results

The combinational circuit that I have designed has 2 inputs and 1 output, Inputs are A and B, and the output is A’(A+B). Transferring this circuit to Vivado, I obtained a scheme of the circuit (Figure: 2). Then writing the testbench code, I get the waveform (Figure: 3) of the circuit and checked it with the truth table of my circuit (Table: 1). After that I attatched the BASYS3 to the computer so as to demonstrate the combinational logic circuit on it. After writing the constraint file (Figure: 1), which is assigning the inputs and outputs on the BASYS3, I tried all 4 possibility by using the switches, and observed the change in the LED light (Figure: 8,9,10,11).

Figure: 2 RTL Schematic

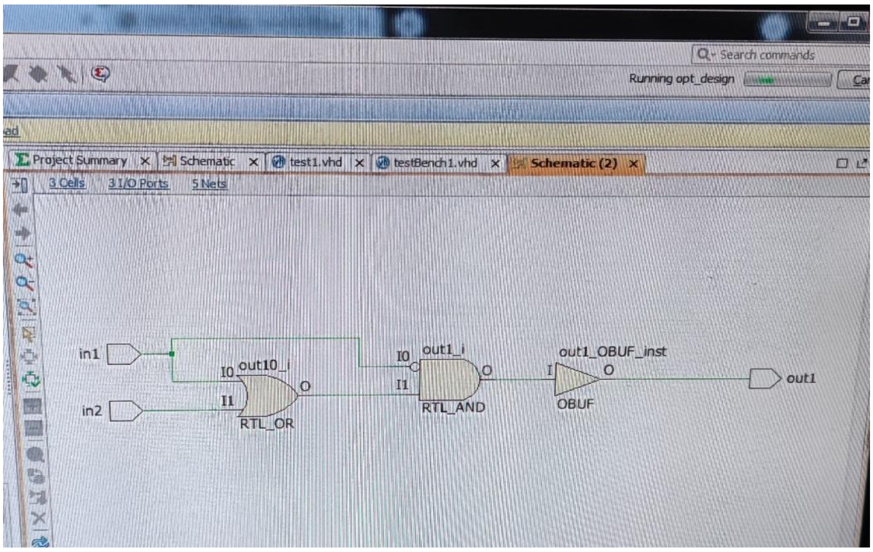


Table: 1 Truth table of the circuit

A (input 1)	B (input 2)	A’(A+B) (output)
0	0	0
0	1	1
1	0	0
1	1	0

Figure: 4,5,6,7 Waveform of the logic circuit

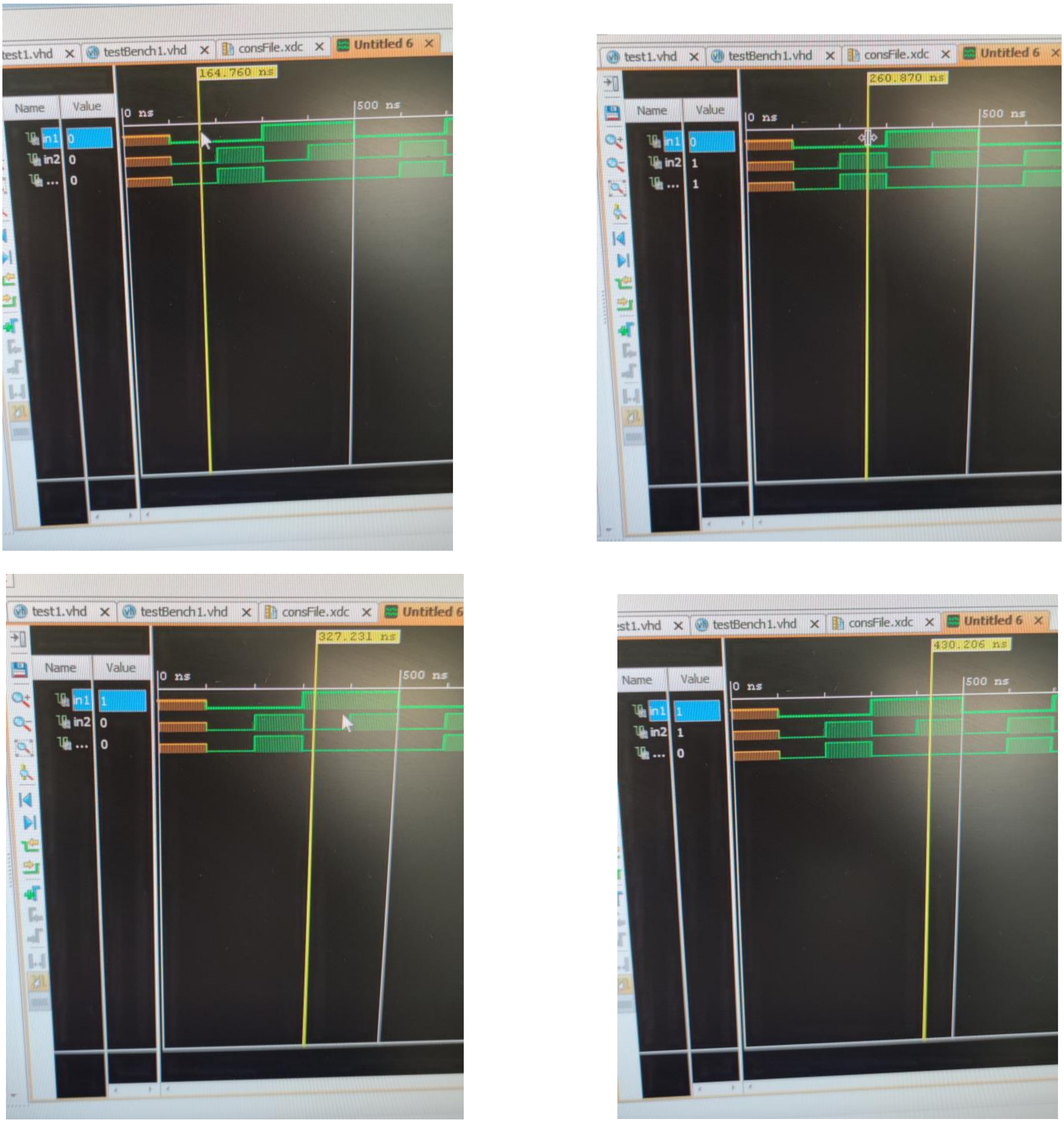
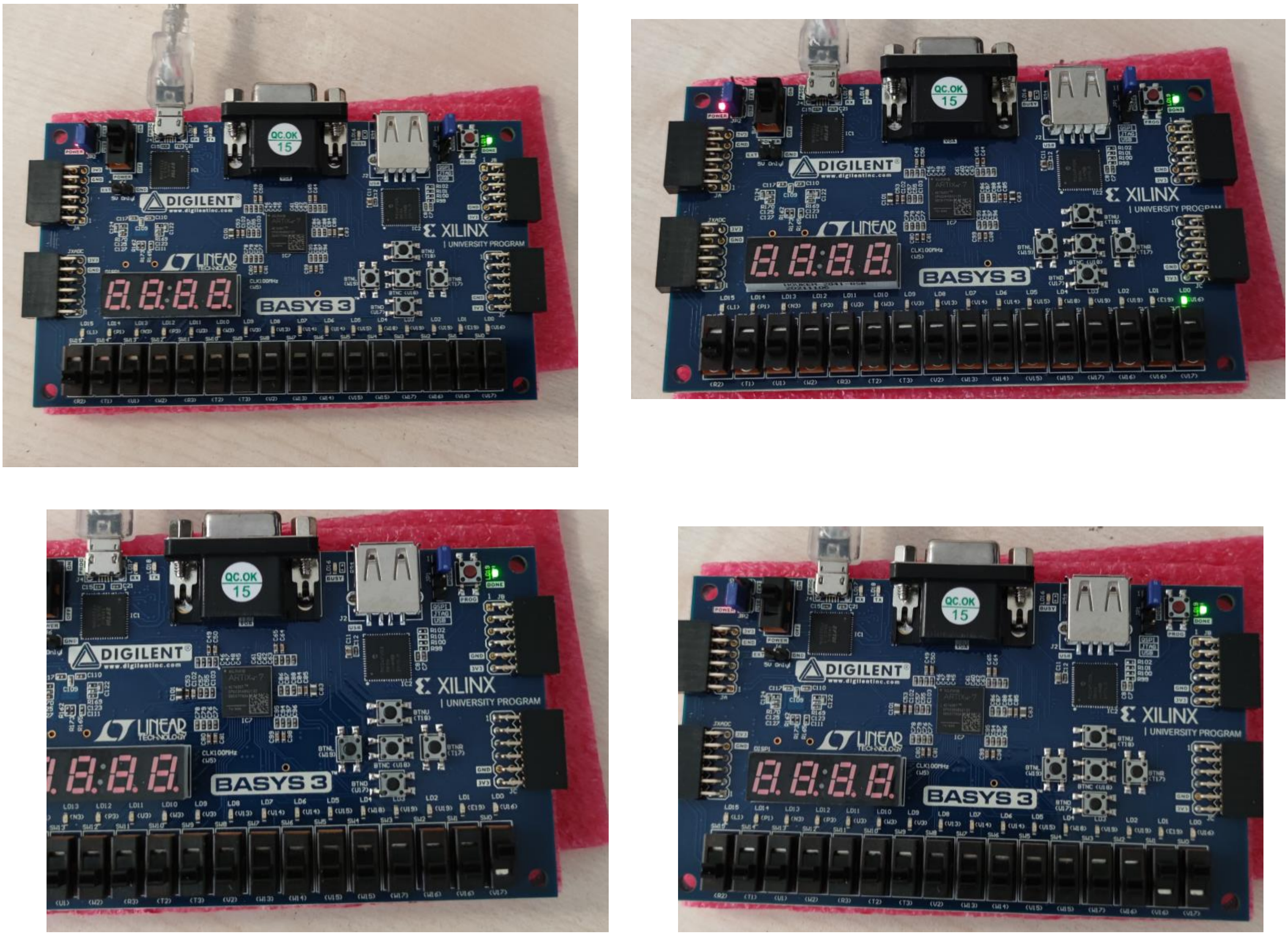


Figure: 8,9,10,11 Demonstration on BASYS3 FPGA



Conclusion

The purpose of the experiment is implementing a logical circuit to Vivado and then demonstrating it on BASYS3 FPGA. The experiment is overall successful. I learned how to use the switches and observe the change on the LED lights.

Appendices

VHDL Codes

```
entity test1 is
        Port ( in1 : in STD_LOGIC;
              in2 : in STD_LOGIC;
              out1 : in STD_LOGIC);
end test1;
architecture Behavioral of test1 is
begin
out1 <= not in1 and (in1 or in2);
----TESTBENCH----
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
entity testBench1 is
end testBench1;
architecture Behavioral of testBench1 is
COMPONENT test1
 PORT( in1 : IN STD_LOGIC;
         in2 : IN STD_LOGIC;
         out1 : OUT STD_LOGIC);
 END COMPONENT;
 SIGNAL in1 : STD_LOGIC;
 SIGNAL in2 : STD_LOGIC;
 SIGNAL out1 : STD_LOGIC;
```

```
BEGIN
 UUT: test1 PORT MAP(
            in1 => in1,
            in2 => in2,
            out1 => out1
 );
 testBench1 : PROCESS
 BEGIN
 wait for 100 ns;
 in1<='0';
 in2<='0';
wait for 100 ns;
 in1<='0';
 in2<='1';
wait for 100 ns;
 in1<='1';
 in2<='0';
 wait for 100 ns;
 in1<='1';
 in2<='1';
 END PROCESS;
end Behavioral;
```

(Taken from BASYS 3 - Vivado Tutorial file)