An-Najah National University
Department of Computer Engineering
Digital design 2 LAB -10636391-
Second Semester 2020/2021

جامعة النجاح الوطنية
قسم هندسة الحاسوب
مختبر تصميم دوائر الكترونية 2 - 10636391-
الفصل الثاني 2021/2020

Instructor: Anas Toma

NAME	ID NUM	SEC
1- أحمد عبد الله جبر القرم	11819195	3/8:00-11:00
-	-	-

Experiment 6: IP cores (Multiplier)

Introduction :

In this lab I will learn how to use and integrate IP cores in my design.

Objectives:

- Lern how to integrate IP cores (a Multiplier).

Tools used in Lap :

1- Computer lap.

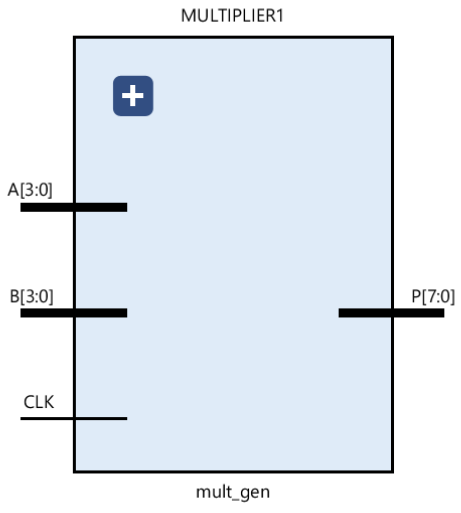
2- Vivado software .

3- Zboard from Xilinx.

4- VHDL

Procedure :

I had followed the Lab Procedure in manual and Instantiated IP into my main Design as shown :



Project Summary | Main.vhd *

E:/projects/IP_coresMultiplier/IP_coresMultiplier.srcs/sources_1/new/Main.vhd

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  entity Main is
4      Port (
5          CLK : in STD_LOGIC;
6          A : in STD_LOGIC_VECTOR (3 downto 0);
7          B : in STD_LOGIC_VECTOR (3 downto 0);
8          P : out STD_LOGIC_VECTOR (7 downto 0)
9      );
10 end Main;
11 architecture Behavioral of Main is
12 COMPONENT mult_gen
13   PORT (
14     CLK : IN STD_LOGIC;
15     A : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
16     B : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
17     P : OUT STD_LOGIC_VECTOR(7 DOWNTO 0)
18   );
19 END COMPONENT;
20 begin
21 MULTIPLIER1 : mult_gen
22   PORT MAP (
23     CLK => CLK,
24     A => A,
25     B => B,
26     P => P
27   );
28 end Behavioral;
```

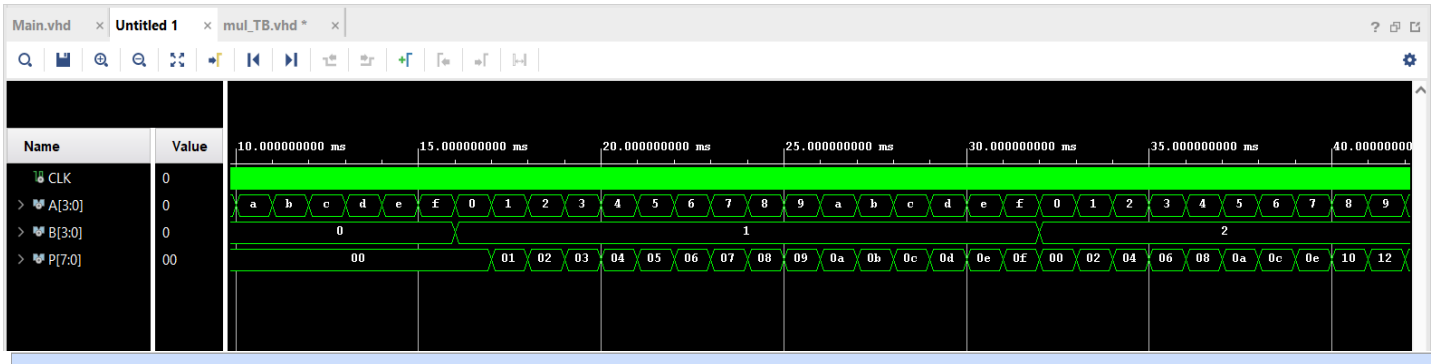
Simulation Code TestBench :

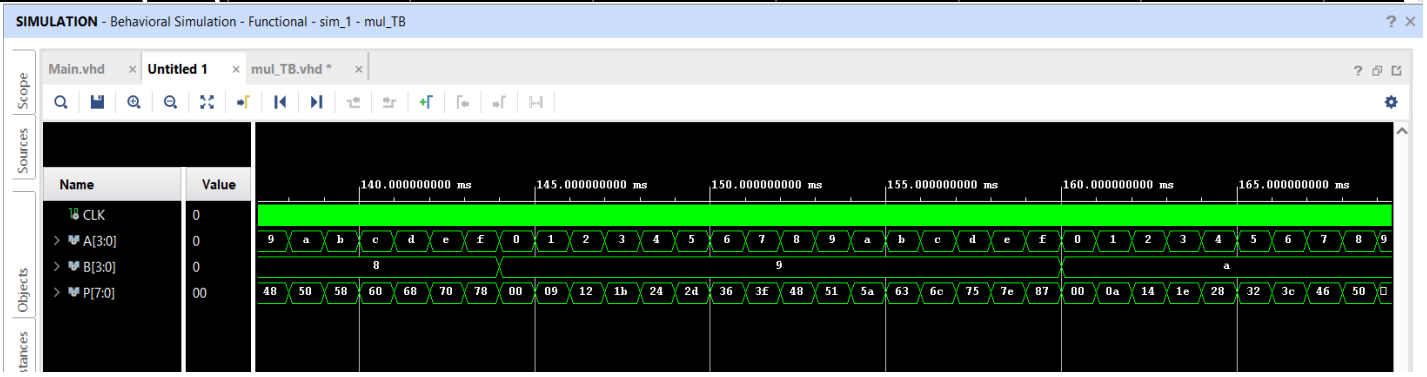
Main.vhd | Untitled 1 | mul_TB.vhd *

E:/projects/IP_coresMultiplier/IP_coresMultiplier.srcs/sim_1/new/mul_TB.vhd

```
 8   architecture Behavioral of mul_TB is
 9   component Main ...
17   signal CLK:std_logic :='0';
18   signal A,B:std_logic_vector (3 downto 0):="0000";
19   signal P:std_logic_vector (7 downto 0);
20   begin
21   UUT:Main port map (
22       CLK => CLK ,
23       A => A ,
24       B => B ,
25       P => P
26       );
27   CLK_G:process
28   begin
29       CLK<=not CLK;
30       wait for 5 ns;
31   end process;
32   process
33   begin
34       wait for 1 ms;
35       A<=A+1;
36       if (A="1111")then
37           A<="0000";
38           B<=B+1;
39       end if;
40       if (B="1111" and A="1111")then
41           wait;
42       end if;
43
44   end process;
45   end Behavioral;
46
```

Simulation:





Synthesis, Implementation, and Bitstream Generation :

I had connected Pins in the board as shown in the Table

Inputs	Package Pin	I/O Std
SW0	F22	LVCMOS18
SW1	G22	LVCMOS18
SW2	H22	LVCMOS18
SW3	F21	LVCMOS18
SW4	H19	LVCMOS18
SW5	H18	LVCMOS18
SW6	H17	LVCMOS18
SW7	M15	LVCMOS18
Outputs	**Package Pin**	**I/O Std**
LD0	T22	LVCMOS33
LD1	T21	LVCMOS33
LD2	U22	LVCMOS33
LD3	U21	LVCMOS33
LD4	V22	LVCMOS33
LD5	W22	LVCMOS33
LD6	U19	LVCMOS33
LD7	U14	LVCMOS33

```
set_property IOSTANDARD LVCMOS18 [get_ports {A[3]}]
set_property IOSTANDARD LVCMOS18 [get_ports {A[2]}]
set_property IOSTANDARD LVCMOS18 [get_ports {A[1]}]
set_property IOSTANDARD LVCMOS18 [get_ports {A[0]}]
set_property IOSTANDARD LVCMOS18 [get_ports {B[3]}]
set_property IOSTANDARD LVCMOS18 [get_ports {B[2]}]
set_property IOSTANDARD LVCMOS18 [get_ports {B[1]}]
set_property IOSTANDARD LVCMOS18 [get_ports {B[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports CLK]
set_property PACKAGE_PIN Y9 [get_ports CLK]
set_property IOSTANDARD LVCMOS33 [get_ports {P[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {P[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {P[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {P[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {P[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {P[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {P[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {P[0]}]
set_property PACKAGE_PIN U14 [get_ports {P[7]}]
set_property PACKAGE_PIN U19 [get_ports {P[6]}]
set_property PACKAGE_PIN W22 [get_ports {P[5]}]
set_property PACKAGE_PIN V22 [get_ports {P[4]}]
set_property PACKAGE_PIN U21 [get_ports {P[3]}]
set_property PACKAGE_PIN U22 [get_ports {P[2]}]
set_property PACKAGE_PIN T21 [get_ports {P[1]}]
set_property PACKAGE_PIN T22 [get_ports {P[0]}]
set_property PACKAGE_PIN F21 [get_ports {A[3]}]
set_property PACKAGE_PIN H22 [get_ports {A[2]}]
set_property PACKAGE_PIN G22 [get_ports {A[1]}]
set_property PACKAGE_PIN F22 [get_ports {A[0]}]
set_property PACKAGE_PIN M15 [get_ports {B[0]}]
set_property PACKAGE_PIN H17 [get_ports {B[1]}]
set_property PACKAGE_PIN H18 [get_ports {B[2]}]
set_property PACKAGE_PIN H19 [get_ports {B[3]}]
```

Conclusion:

In this experiment, I learned how to deal with IP cores .And how to integrate IP cores in my design .Also I'm now aware how much is IP core subject important in reduce cost ,time ,and efforts .