

Experiment 1

Combinational Circuits - 1

Part 1

Mayur Ware, 19D070070
EE-214, WEL, IIT Bombay

February 15, 2021

Overview of the experiment

- The purpose of the experiment is to design a block which checks if the given 4 bit number is a BCD or not .
- I used a K-Map to make a equation to directly check if the given number is BCD. And then did some arithmetic manipulations to directly get the output.
- The report contains a handmade diagram which describes my approach, some important extracts of VHDL code, the output waveforms and Krypton Observations.

Approach to the experiment

I used a K-Map to make a equation to directly check if the given number is BCD.

The equation turns out to be,

$$Y = \overline{A_3} + \overline{A_2} \cdot \overline{A_1}$$

Then I did bitwise OR with every bit of given 4 bit number with \overline{Y} which directly gives the output.

<u>BCD Identifier</u>					
Sr.	A ₃	A ₂	A ₁	A ₀	Y
1	0	0	0	0	1
2	0	0	0	1	1
3	0	0	1	0	1
4	0	0	1	1	1
5	0	1	0	0	1
6	0	1	0	1	1
7	0	1	1	0	1
8	0	1	1	1	1
9	1	0	0	0	1
10	1	0	0	1	0
11	1	0	1	0	0
12	1	0	1	1	0
13	1	1	0	0	0
14	1	1	0	1	0
15	1	1	1	0	0
16	1	1	1	1	0

Here, \overline{Y} is most significant bit

$(\overline{A_3} \cdot \overline{A_2} \cdot \overline{A_1} \cdot \overline{A_0}) + \overline{Y} = \text{output}$

$\text{Output}(3) = A_3 + \overline{Y}$

$\text{Output}(2) = A_2 + \overline{Y}$

$\text{Output}(1) = A_1 + \overline{Y}$

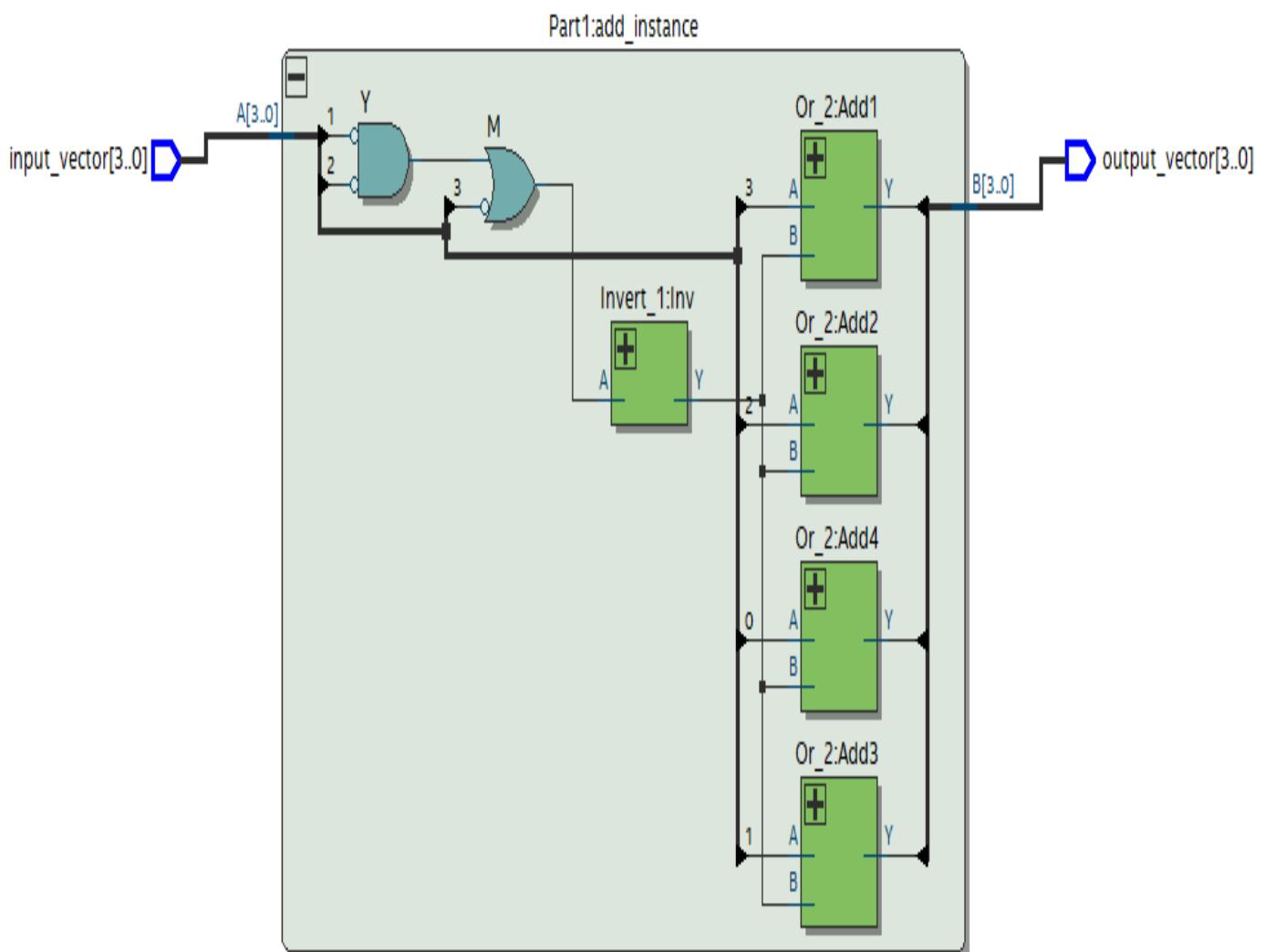
$\text{Output}(0) = A_0 + \overline{Y}$

Design document and VHDL code if relevant

Main Entity :

```
entity Part1 is
port (A : in std_logic_vector(3 downto 0);
      B : out std_logic_vector(3 downto 0));
end entity Part1;
architecture Struct of Part1 is
signal Y, M, S: std_logic;
begin
Y <= (not A(3) or ((not A(2)) and (not A(1))));
M <= Y;
begin
Inv : Invert_1 port map ( A => M, Y => S);
Add1 : Or_2 port map ( A=> A(3), B => S, Y => B(3));
Add2 : Or_2 port map ( A=> A(2), B => S, Y => B(2));
Add3 : Or_2 port map ( A=> A(1), B => S, Y => B(1));
Add4 : Or_2 port map ( A=> A(0), B => S, Y => B(0));
end Struct;
```

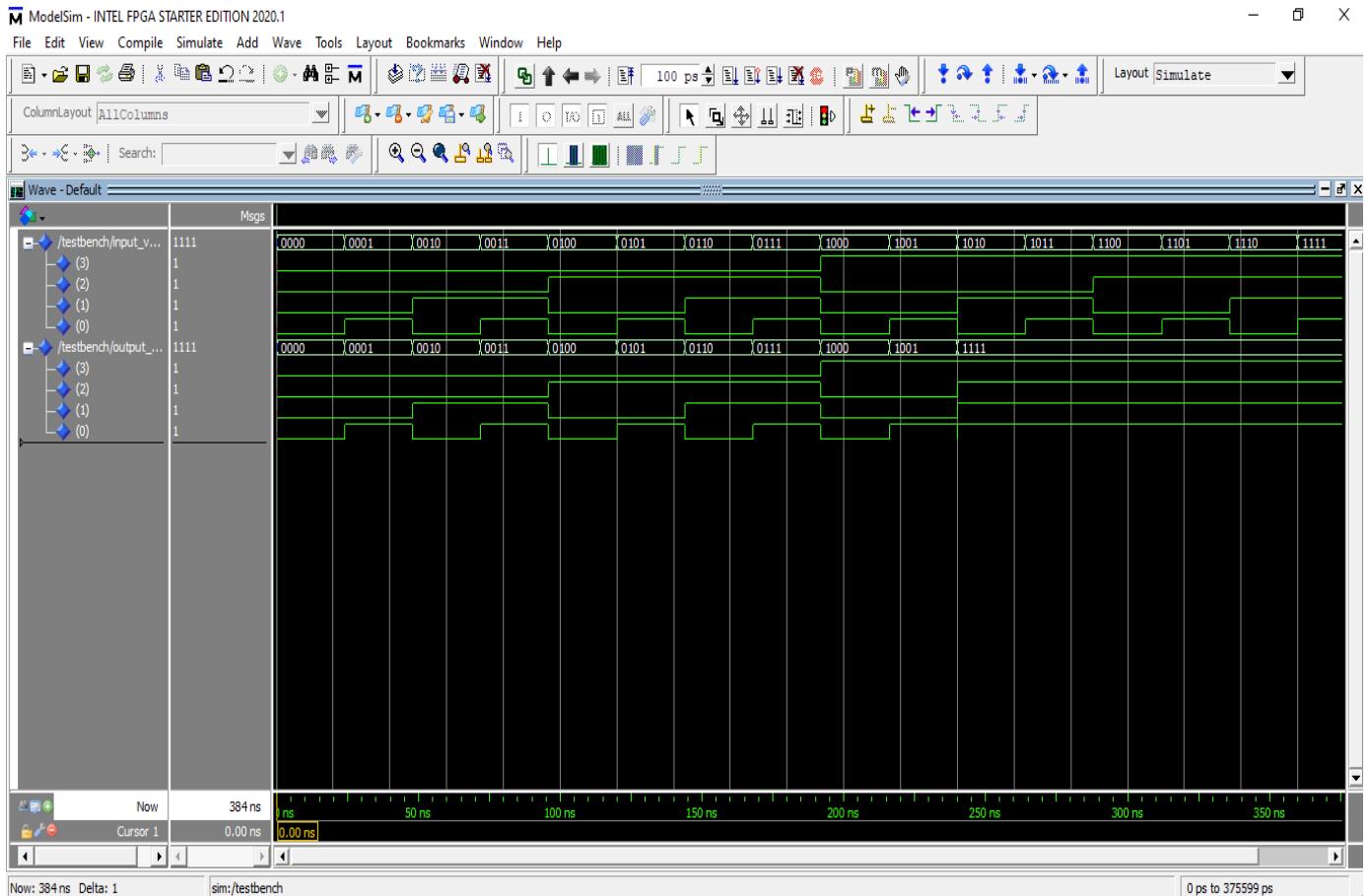
RTL View



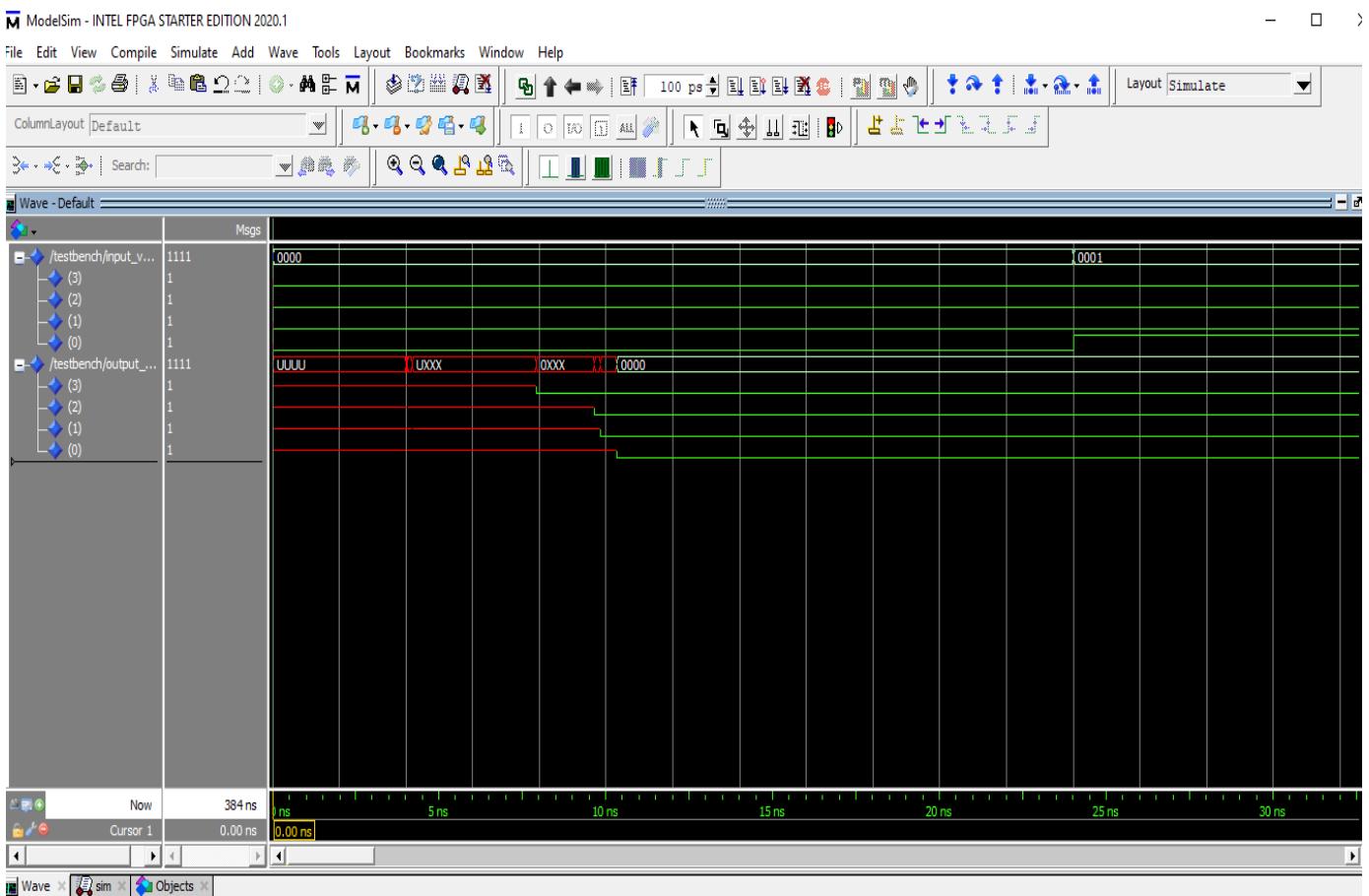
DUT Input/Output Format

A : in std_logic_vector(3 downto 0);	-Input
B : out std_logic_vector(3 downto 0));	-Output
0111 0111 1111	-Input Bits, Output Bits, Mask Bits

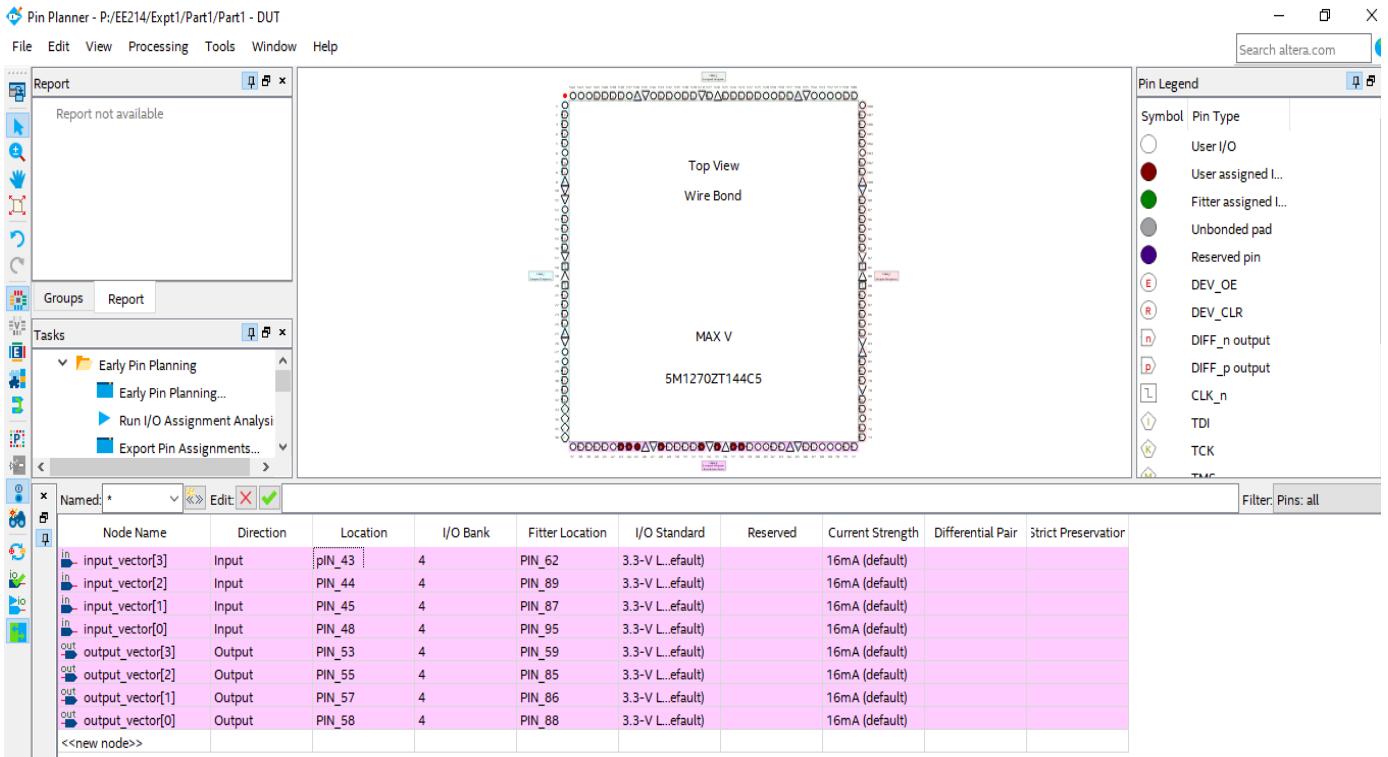
RTL Simulation



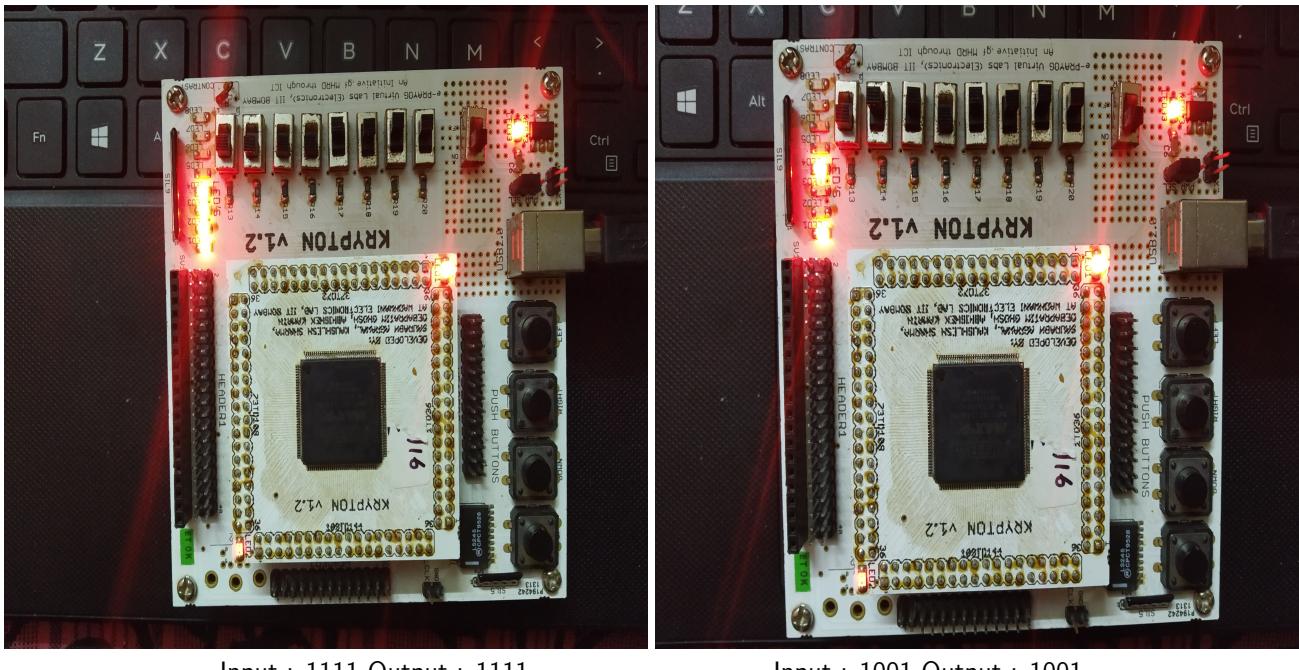
Gate-level Simulation



Krypton board



Pin Planner



Observations

All the recorded observations are :

Input	Expected Output	Observed Output
0100	0100	0100
0110	0110	0110
1100	1111	1111
1011	1111	1111

Tiva C

After generating all 16 testcases in in.txt, Tiva-C compiled outputs in output.txt.
Image attached below shows the successful compilation.

```
#----- Command - 28 : RUNTEST 1 MSEC -----#
#
#----- Command - 29 : SDR 4 TDI(E) 4 TDO(F) MASK(F) -----#
Successfully entered the input..
Sampling out data..
F
Output Comparison : Success

#----- Command - 30 : RUNTEST 1 MSEC -----#
#----- Command - 31 : SDR 4 TDI(F) 4 TDO(F) MASK(F) -----#
Successfully entered the input..
Sampling out data..
F
Output Comparison : Success

#----- Command - 32 : RUNTEST 1 MSEC -----#
Sampling out data..
F
Output Comparison : Success
OK. All Test Cases Passed.
Transaction Complete.

P:\EE214\Expt1\Part1>
```

References

EE 309 : Microprocessors by Prof. Dinesh Sharma Video Lectures for understanding the logic and design of BCD.