Experiment -1: Write a VHDL code for all the logic gates

Objective:

The objective of this experiment is to:

i. To revise the working of various logic gates

ii. To learn the VHDL coding

iii. To simulate for functional verification

iv. To implement on CPLD / FPGA

1. TITLE: AND gate

TRUTH TABLE:

x	y	z
0	0	0
0	1	0
1	0	0
1	1	1

Logic Equation:

Z = X.Y

This equation can be used for describing the dataflow model of AND architecture

We observe that the output is high when both X and Y are high ‘1’, otherwise the output is low ‘0’. The property can be used in modeling the sequential behaviour

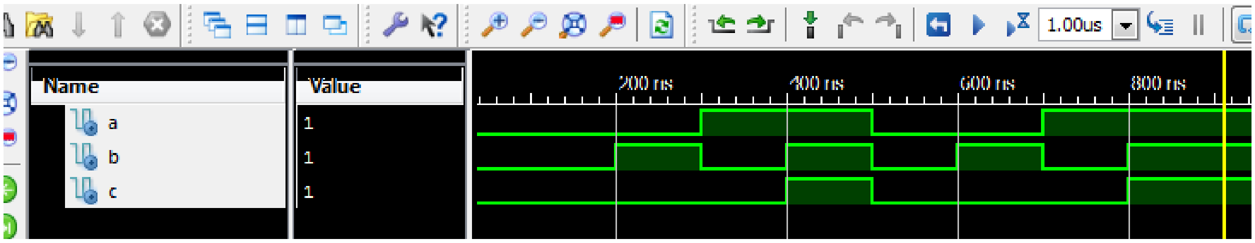
VHDL CODE:

```
Library IEEE;
use IEEE.std_logic_1164.all;
entity AND2 is
        port(
                A : in STD_LOGIC;
                B : in STD_LOGIC;
                C : out STD_LOGIC
```

```
                    );
          end AND2;

-- The three architectural models are given below:
--1.      Dataflow model
          architecture  behav1  of  AND2  is
          begin
            C <= A and B;        --Signal Assignment Statement
          end behav1;
-- 2.     Behavioral model
          architecture  behav2  of  AND2  is
          begin
            process  (A,  B)
            begin
                if (A='1' and B='1') then
                  C<= '1';
                else
                  C<= '0';
                end if;
            end process; end behav2;
```

OUT PUT WAVE FORM:



Name	Value
a	1
b	1
c	1

2. TITLE: OR gate

TRUTH TABLE:

x	y	z
0	0	0
0	1	1
1	0	1
1	1	1

Logic Equation:

i. **Z = X + Y This can be used for describinbg the dataflow model of AND architecture**

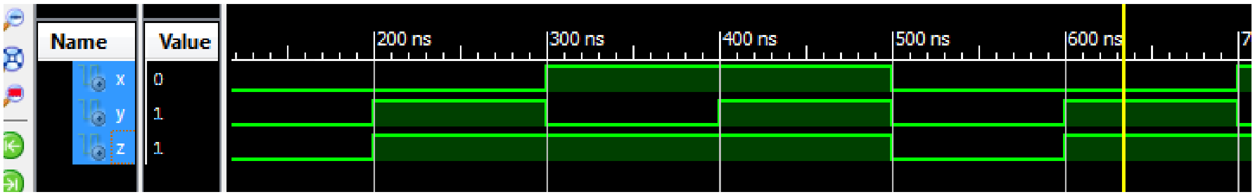
ii. **We observe that the output is high when any or all the inputs X and Y are high '1', otherwise the output is low '0'. The property can be used in modeling the sequential behaviour**

VHDL CODE

```
 Library IEEE;
use IEEE.std_logic_1164.all;
 entity OR2 is
          port(
                   x : in STD_LOGIC;
                   y : in STD_LOGIC;
                   z : out STD_LOGIC
               );
 end OR2;
 --Dataflow model
 architecture  behav1  of  OR2  is
 begin
    Z <= x or y;          --Signal Assignment Statement
 end behav1;
-- Behavioral model
 architecture  behav2  of  OR2  is
```

```
begin
    process (x, y)
    begin
          if (x='0' and y='0') then
            Z <= '0';
         else
            Z<= '1';
         end if;
     end process; end behav2;
```

OUTPUT WAVEFORM:



Name	Value
x	0
y	1
z	1

3. TITLE: NOT gate

TRUTH TABLE:

x	z
0	1
1	0

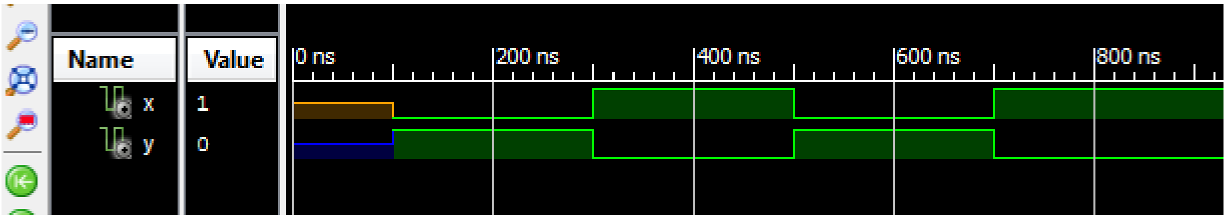
1. **Y = X’ This can be used for describing the dataflow model of AND architecture**
2. **We observe that the output is high when inputs X low ‘0’, and the output is low when input X is high ‘1’. The property can be used in modeling the sequential behavior**

VHDL CODE:

```
        Library IEEE;
       use IEEE.std_logic_1164.all;
        entity not1 is
                port(
                        X:    in  STD_LOGIC;
                      Y:      out   STD_LOGIC
                      );
                end not1;
--Dataflow model
        architecture  behav1  of  not1  is
        begin
          Y<= not X;   --Signal   Assignment   Statement
        end behav1;
-- Behavioral model
        architecture  behav2  of  not1  is
        begin
            process (X)
```

```
                begin
                       if (x='0') then  -- Compare with truth table
                       Y<= '1';
                      else
                       Y<= '0';
                      end if;
                      end process;
            end behav2;
```

OUTPUT WAVEFORM for NOT Gate:



4. TITLE: NAND gate

TRUTH TABLE:

x	y	z
0	0	1
0	1	1
1	0	1
1	1	0

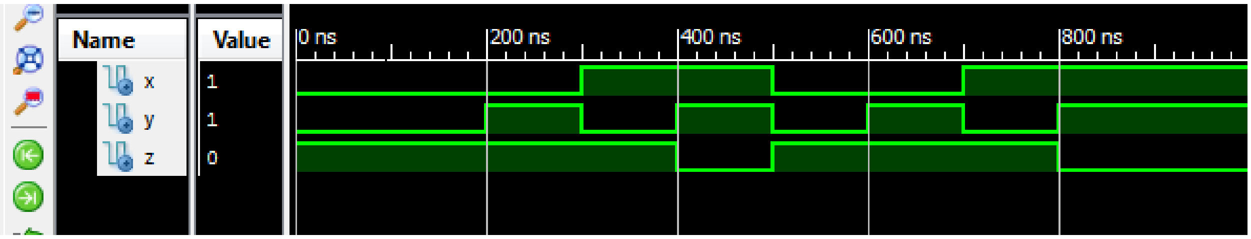
i. **Y = (X.Y)'   This can be used for describing the dataflow model of AND architecture**

ii. **We observe that the output is high when any or all inputs X and Y are low '0', and the output is low when all input are high '1'. The property can be used in modeling the sequential behavior**

VHDL CODE:

```
    Library IEEE;
   use IEEE.std_logic_1164.all;
    entity nand2 is
              port(
                       x : in STD_LOGIC;
                       y : in STD_LOGIC;
                       z : out STD_LOGIC
                 );
    end nand2;
--Dataflow model
    architecture  behav1  of  nand2  is
    begin
             z<= x nand y;            --Signal Assignment Statement
    end behav1;
 -- Behavioral model
    architecture  behav2  of  nand2  is
    begin
     Process   (x,   y)
     Begin
          If (x='1' and y='1') then
               Z <= '0';
          else
               Z <= '1';
          end if;
     end process; end behav2;
```

OUTPUT WAVEFORM



5. **TITLE**: NOR gate

TRUTH TABLE:

x	y	z
0	0	1
0	1	0
1	0	0
1	1	0

i. **Y = (X + Y)' This can be used for describing the dataflow model of AND architecture**

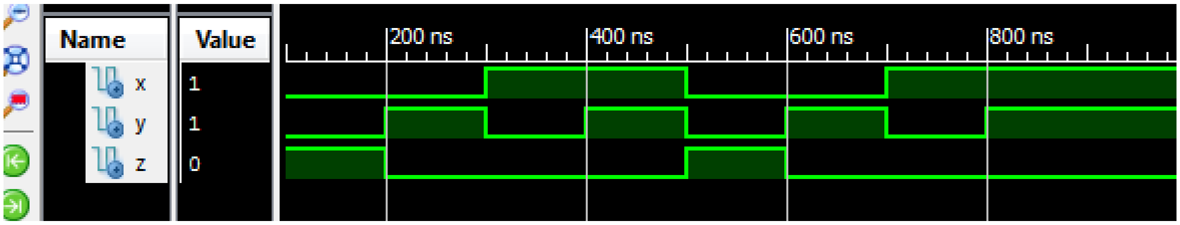
ii. **We observe that the output is high when all inputs X and Y are low '0', and the output is low when any or all input are high '1'. The property can be used in modeling the sequential behavior**

VHDL CODE:

```
            Library IEEE;
           use IEEE.std_logic_1164.all;
            entity nor2 is
                         Port (
                                    X: in STD_LOGIC;
                                    Y: in STD_LOGIC;
                                    Z: out STD_LOGIC
                            );
            end nor2;
--Dataflow model
            architecture  behav1  of  nor2  is
            begin
            Z<= x nor y;
            end behav1;
-- Behavioral model
architecture behav2 of nor2 is begin
```

```
                process (x, y)
                 begin
--Signal Assignment Statement
                   If (x='0' and y='0') then
                     Z <= '1';
                   else
                     Z <= '0';
                   end if;
                end process;
              end behav2;
```

OUTPUT WAVEFORM:



Name	Value
x	1
y	1
z	0

6. TITLE:EX-OR gate

TRUTH TABLE:

a. **Y = (X ‘.Y +X .Y’) This can be used for describing the dataflow model of AND architecture**

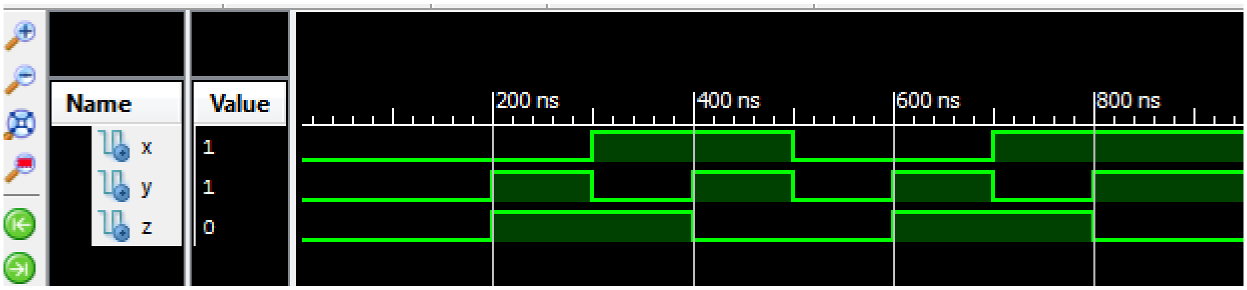
b. **We observe that the output is low when all inputs X and Y are low ‘0’ or when all the inputs are high. The output is high when any input is high ‘1’. The property can be used in modeling the sequential behavior**

x	y	z
0	0	0
0	1	1
1	0	1
1	1	0

VHDL CODE:

```
Library IEEE;
use IEEE.std_logic_1164.all;
    entity xor2 is
                Port ( X: in STD_LOGIC;
                            Y: in STD_LOGIC;
                            Z: out STD_LOGIC
                );
    end xor2;
--Dataflow model
    architecturedataflow   of   xor2   is
    begin
                Z<= x xor y;                --Signal Assignment Statement
    enddataflow;
--behaviour modelling
            architecture  behav2  of  xor2  is
            begin
                process   (x,   y)
                begin
                      If (x/=y) then
                          Z <= '1';
                       else
                           Z<= '0';
                      end if;
             end process;
end behav2;
```

OUTPUT WAVEFORM


Name
Value
x
y
z
1
1
0
200 ns
400 ns
600 ns
800 ns