EXPERIMENT No.: - 7

Simulation of VHDL Models.

TITLE:

Implementation of AND – OR inverter in VHDL using Structural Modelling.

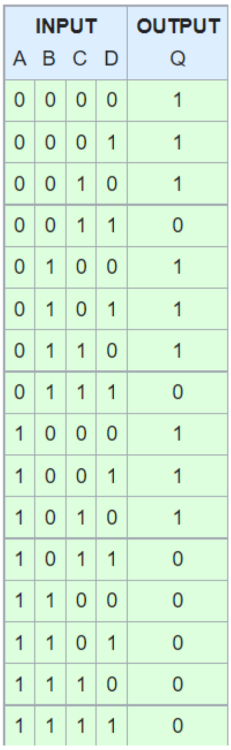
OBJECTIVE:

To implement the Structural model of AND OR inverter using ISE Simulator.

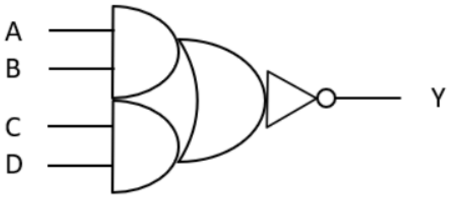
THEORY:

AND-OR-INVERT(AOI) logic and AOI gates are two-level compound (or complex) logic functions constructed from the combination of one or more AND gates followed by a OR and NOT gates.

In digital logic, an inverter or NOT gate is a logic gate which implements logical negation. Truth table for 2-2 AOI:



INPUT				OUTPUT
A	B	C	D	Q
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0



AOI (AND-OR-Inverter) Cell

<u>Structural Modelling:</u> In this modelling, an entity is described as a set of interconnected components. A component instantiation statement is a concurrent statement. Therefore, the order of these statements is not important. The structural style of modelling describes only an interconnection of components (viewed as black boxes), without implying any behaviour of the components themselves nor of the entity that they collectively represent.

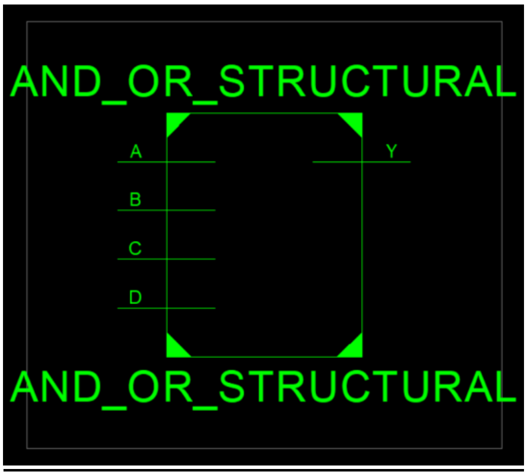
SOURCE CODE: -

```
 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 primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity AND_OR_STRUCTURAL is
   Port ( A : in  STD_LOGIC;
          B : in  STD_LOGIC;
          C : in  STD_LOGIC;
          D : in  STD_LOGIC;
          Y : out  STD_LOGIC);
end AND_OR_STRUCTURAL;

architecture Structural  of AND_OR_STRUCTURAL is

Component AND_g is
Port(P,Q : IN STD_logic;
                        O1 : OUT STD_logic);
```

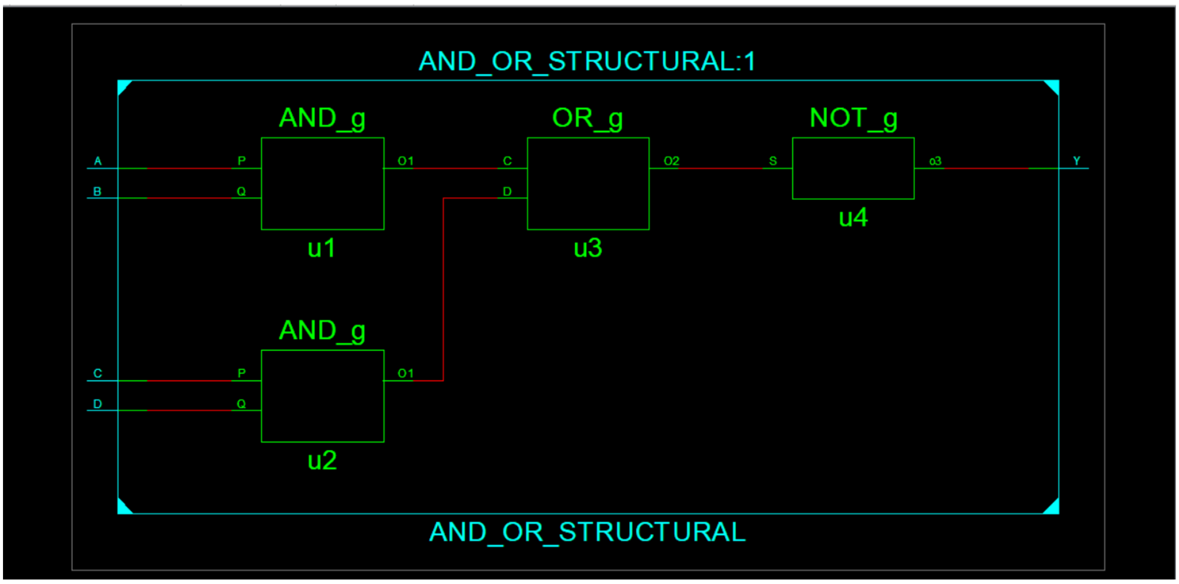
```
end Component;


Component OR_g is
Port(C,D : in STD_LOGIC;
    O2 : out STD_LOGIC);
end Component;


Component NOT_g is
Port( S : in STD_LOGIC;
    o3 : out STD_LOGIC);
end Component;


SIGNAL S1,S2,S3 : STD_LOGIC;


begin
u1 : AND_g PORT MAP(A,B,S1);
u2 : AND_g PORT MAP(C,D,S2);
u3 : OR_g PORT MAP(S1,S2,S3);
u4 : NOT_g PORT MAP(S3,Y);

end Structural ;
```

OUTPUT: -

ENTITY DIAGRAM: -



DETAILED ENTITY DIAGRAM: -



RESULT:

Thus, through structural modelling we have implemented AND-OR-Inverter and we have processed the overview of the basic structure of this entity as a whole using the RTL schematics. The blank square correctly represents the different logic gates, i.e., AND, OR and NOT gates with their respective input output Ports.

CONCLUSION:

In this experiment, we learnt about how to implement the overall structural ports design and placement of the respective logic gates of an AND-OR-Inverter using structural modelling.