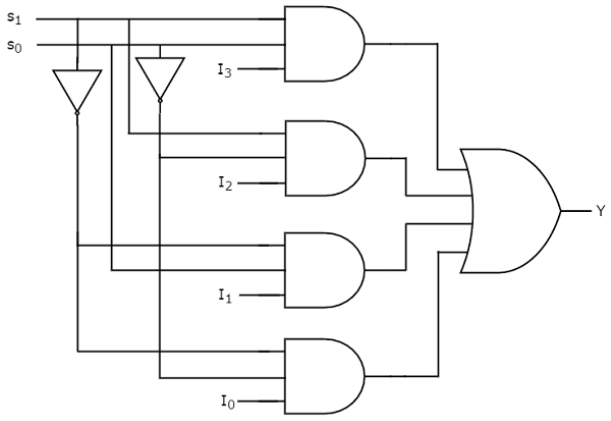
EXPERIMENT No-6

Aim:- To Design a Multiplexer and Demultiplexer using VHDL

Theory:

Multiplexer:

4:1



Selection Lines		**Output**
S_1	**S_0**	**Y**
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

From Truth table, we can directly write the **Boolean function** for output, Y as

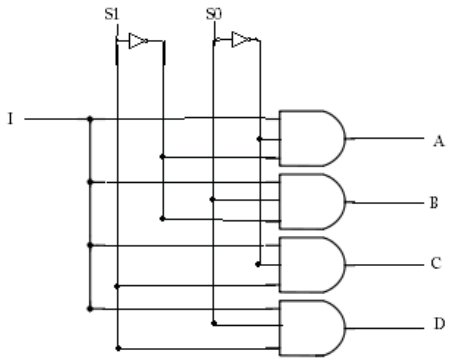
$Y=S_1'S_0'I_0+S_1'S_0\ I_1+S_1\ S_0\ 'I_2+S_1\ S_0\ I_3$

Demultiplexer:

Demultiplexers take one data input and a number of selection inputs, and they have several outputs.

1:4

I	S 1	S 0	ABC D
0	X	X	0000
1	0	0	1000
1	0	1	0100
1	1	0	0010
1	1	1	0001



Program for MUX:

```
library ieee;
use ieee.std_logic_1164.all;
entity Multiplexer is
port(a,b,c,d:in std_logic;
s:in std_logic_vector(1 downto 0);
y:out std_logic);
end Multiplexer;
architecture beh of Multiplexer is
begin
process(a,b,c,d,s)
begin
case s is
when "00"=>y<=a;
when "01"=>y<=b;
when "10"=>y<=c;
when "11"=>y<=d;
when others=>y<='U';
end case;
end process;
end beh;
```

Output:

Program for DEMUX:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity demultiplexer is
Port ( i : in STD_LOGIC;
s0 : in STD_LOGIC;
s1 : in STD_LOGIC;
a : out STD_LOGIC;
b : out STD_LOGIC;
c : out STD_LOGIC;
d : out STD_LOGIC);
end demultiplexer;
architecture Behavioral of demultiplexer is
signal p,q : STD_LOGIC;
```

```
begin
p <= not s0;
q <= not s1;
a<= i and p and q;
b<= i and q and s0;
c<= i and p and s1;
d<= i and s1 and s0;
end Behavioral;
```

……..

```
library IEEE;

use IEEE.STD_LOGIC_1164.ALL;

use IEEE.STD_LOGIC_ARITH.ALL;

use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity DEMUX is

Port ( I : in  STD_LOGIC;

S : in STD_LOGIC_VECTOR (1 downto 0);

Y : out STD_LOGIC_VECTOR (3 downto 0));

end DEMUX;

architecture Behavioral of DEMUX is

begin

process (I, S)

begin

if (S <= "00") then

Y(0) <= I ;
```

```
elsif (S <= "01") then

Y(1) <= I ;

elsif (S <= "10") then

Y(2) <= I ;

else

Y(3) <= I ;

end if;

end process;

end Behavioral;
```

Output: