EXPERIMENT No-5

Aim: To Design a Half Adder and Full Adder using VHDL

Description:

Half adder

Half adder is a combinational arithmetic circuit that adds two numbers and produces a sum bit (S) and carry bit (C) as the output. If A and B are the input bits, then sum bit (S) is the X-OR of A and B and the carry bit (C) will be the AND of A and B.

The sum and carry are as follows:

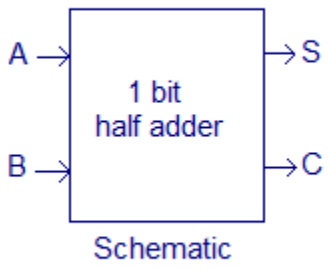
Sum = A'B+AB'= A $\oplus$ B

Carry = AB

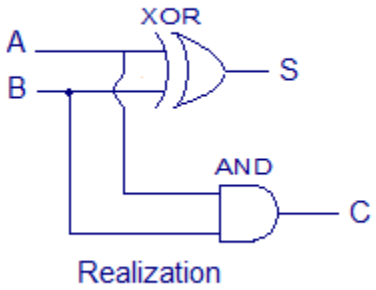
The truth table, schematic representation and XOR//AND realization of a half adder are shown in the figure below.

Inputs		Outputs	
A	B	S	C
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

Truth table



Schematic



Realization

Full adder

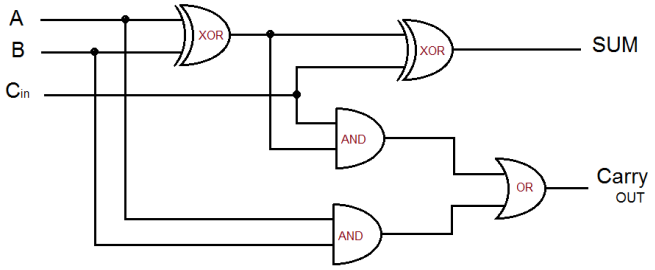
The full adder is used to add three 1-bit binary numbers A, B, and carry C. The full adder has three input states and two output states i.e., sum and carry.

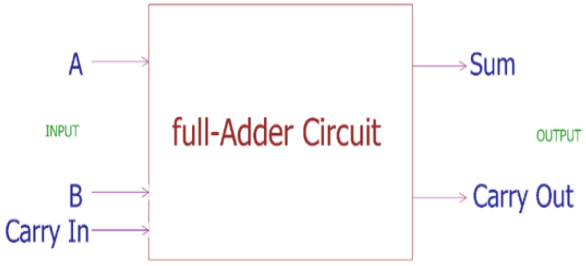
Sum:

- Perform the XOR operation of input A and B.
- Perform the XOR operation of the outcome with carry. So, the sum is (A XOR B) XOR C_{in} which is also represented as: $(A \oplus B) \oplus C_{in}$.

Carry:

- Perform the 'AND' operation of input A and B.
- Perform the 'XOR' operation of input A and B.
- Perform the 'OR' operations of both the outputs that come from the previous two steps. So the 'Carry' can be represented as: $A.B + (A \oplus B)$





Inputs			Outputs	
A	**B**	**C_{in}**	**Sum**	**Carry**
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Program:

--Half adder

```
library ieee ;
use ieee.std_logic_1164.all ;
entity half_adder is
port ( a, b : in std_logic ;
s, c : out std_logic ) ;
end half_adder ;
architecture dataflow_half of half_adder is
begin
s <= a xor b;
c <= a and b;
end dataflow_half;
```

Program:

--Full Adder

1. VHDL code for full adder data flow:

```
library ieee ;
use ieee.std_logic_1164.all ;

entity fa is
Port ( a : in std_logic;
b : in std_logic;
cin : in std_logic;
s : out std_logic;
```

```
cout : out std_logic);
end fa;
architecture Behavioral of fa is
begin
s <= (a xor b) xor cin;
cout <= (a and b) or (b and cin) or (a and cin);
end Behavioral;
```

2. VHDL code for full adder behavioral:

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity fa1 is
Port ( a,b,ci : in std_logic;
s,co : out std_logic
);
end fa1;

architecture Behavioral of fa1 is
begin
process (a,b,ci)
begin
s<=a xor b xor ci;
```

```
co<=(a and b)or (b and ci)or (ci and a);

end process;

end Behavioral;
```

Output:

Half Adder:

Full Adder: