VHDL Lab Day 2

Behavioral Model

AND GATE IN BEHAVIOURAL MODEL

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity and_bv is
    Port ( a : in STD_LOGIC;
           b : in STD_LOGIC;
           c : out STD_LOGIC);
end and_bv;
```

AND GATE IN BEHAVIOURAL MODEL

```
architecture Behavioral of and_bv is
begin
   process(a,b)
   begin
      if(a='1')and(b='1')then
         c<='1';
      else
         c<='0';
      end if;
   end process;
end Behavioral;
```

SIMULATION SOURCE

```
entity and_bv_tb is
--    Port ( );
end and_bv_tb;
architecture Behavioral of and_bv_tb is
component and_bv is
    Port ( a : in STD_LOGIC;
           b : in STD_LOGIC;
           c : out STD_LOGIC);
end component;
signal a1 : STD_LOGIC := '0';
signal b1 : STD_LOGIC := '0';
signal c1 : STD_LOGIC;
begin
uut: and_df PORT MAP(a=>a1,b=>b1,c=>c1);
   stim_proc: process
   begin
      wait for 100 ns;
      a1<='0';       b1<='1';
      wait for 100 ns;
      a1<='1';  b1<='0';
      wait for 100 ns;
      a1<='1';  b1<='1';
      wait;
    end process;
end Behavioral;
```

OR GATE IN BEHAVIOURAL MODEL

```
architecture Behavioral of or_bv is
begin
   process(a,b)
   begin
      if(a='1')or(b='1')then
         c <='1';
      else
         c <='0';
      end if;
    end process;
end Behavioral;
```

```
architecture Behavioral of or_bv is
begin
   process(A,B)
   begin
   if(A='0' and B='0')then
      C<='0';
   else
      C<='1';
   end if;
   end process;
end Behavioral;
```

COMBINATIONAL CIRCUITS

- ***HALF ADDER :*** An **adder** is a digital circuit that performs addition of numbers. The **half adder** adds two single binary digits and has two outputs, sum and carry. The carry signal represents an overflow into the next digit of a multi-digit addition.

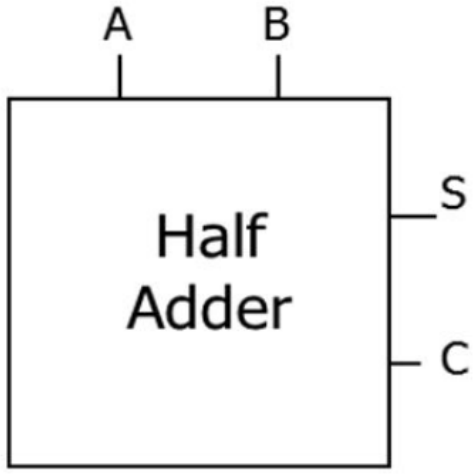

Figure 1: Block diagram of the half adder

A	B	S (sum)	C (carry)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Table 1: Truth table of the half adder.

HALF ADDER

$$S = \overline{A} \bullet B + A \bullet \overline{B} = A \oplus B$$
$$C = A \bullet B$$

```
entity ha_df is
    Port ( a : in STD_LOGIC;
           b : in STD_LOGIC;
           s : out STD_LOGIC;
           c : out STD_LOGIC);
end ha_df;

architecture Behavioral of ha_df is
begin
s <= a xor b;
c <= a and b;
 end Behavioral;
```

```
architecture Behavioral of ha_bv is
 begin
process(a,b)
    begin
     if(a=b)then
       s<='0';
     else
       s<='1';
     end if;
     if(a='1')and(b='1')then
       c<='1';
     else
       c<='0';
     end if;
    end process;
end Behavioral;
```

FULL ADDER : -

- The **full adder** adds three single binary digits and has two outputs - sum and carry.

A	B	Cin	Sum	Cout
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

FULL ADDER

```
architecture Behavioral of fa_df is
begin
 sum <= x xor y xor z;
 carry <= (x and y)or(y and z)or (z and x);
end;
```

```
architecture Behavioral of full_adder_behavioral is
begin
   process(A,B,C)
   begin
   if(A='0')then
            if(B=C) then
                        S<='0’;
             else
                        S<='1';
            end if;
            if(B='1' and C='1')then
                        Cout<='1';
            else
                        Cout<='0';
      end if;     end if;
 if(A='1')then
       if(B=C)then
          S<='1';
       else
          S<='0';
       end if;
       if(B='0' and C='0')then
          Cout<='0';
       else
          Cout<='1';
       end if;
    end if;
end process;
end Behavioral;
```

Assignments-2 Day 2

- Design Behavioural models of following logic gates.
- AND,OR,NAND,NOR,XOR,NOT
- Design Data Flow and Behavioural models of Half adder and Full adder