Institute of Engineering & Management
Department of Computer Science & Engineering
Computer Architecture Laboratory for 2nd year 4th semester 2018
Code: CS 493

Date: 07/04/2018

WEEK-7

Assignment 1: Implementation of Full Adder & Parallel Adder

Objective: Implement full adder using 2 half adder

Software used:

Property Name	Value
Device family	Spartan 3
Device	XC3S50
Package	PQ208
Speed	-5
Top-level source type	HDL
Synthesis Tool	XST(VHDL/Verilog)
Simulator	ISE Simulator
Preferred Language	VHDL

Theory:

i. Make VHDL with required port specification.
ii. Calculate the value c for both data flow and behavioral model using if and else statement with required condition.
iii. Make test bench waveform and check for the given inputs.

Structural Model:

Name: Ranajit Roy **Roll:** 47 **Sec:** A

Code: **HALF ADDER**

```
begin
s<=a xor b;
c<=a and b;
end Behavioral;
```

OR GATE

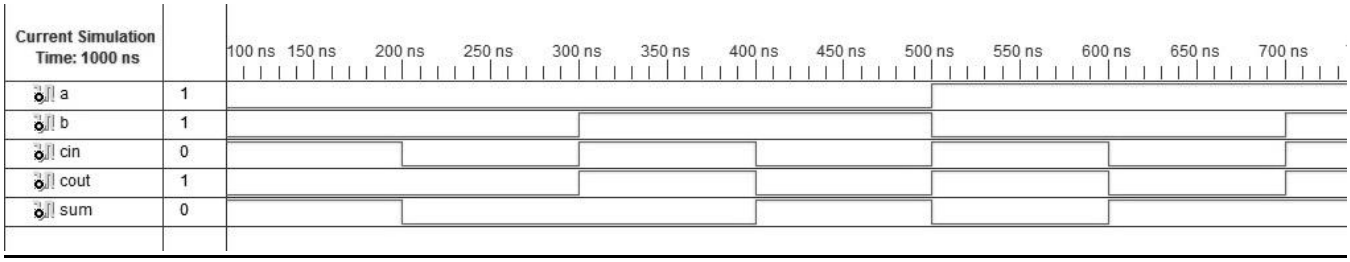
```
begin
c<=a or b;
end Behavioral;
```

FULL ADDER

```
entity fa is
   Port ( a : in  STD_LOGIC;
          b : in  STD_LOGIC;
          cin : in  STD_LOGIC;
          cout : out  STD_LOGIC;
          sum : out  STD_LOGIC);
end fa;
architecture Behavioral of fa is
            COMPONENT ha
            PORT(
                        a : IN std_logic;
                        b : IN std_logic;
                        c : OUT std_logic;
                        s : OUT std_logic
                        );
            END COMPONENT;
                        COMPONENT or1
            PORT(
                        a : IN std_logic;
                        b : IN std_logic;
                        c : OUT std_logic
                        );
            END COMPONENT;
signal x: Std_logic;
signal y: Std_logic;
signal z: Std_logic;

begin
a1: ha port map(a,b,x,y);
a2: ha port map(cin,y,z,sum);
a3: or1 port map(x,z,cout);
end Behavioral;
```

Output:



Name: Ranajit Roy **Roll:** 47 **Sec:** A

Objective: Implement 4 bit Parallel adder using full adder

Software used:

Property Name	Value
Device family	Spartan 3
Device	XC3S50
Package	PQ208
Speed	-5
Top-level source type	HDL
Synthesis Tool	XST(VHDL/Verilog)
Simulator	ISE Simulator
Preferred Language	VHDL

Theory:

i. Make VHDL with required port specification.
ii. Calculate the value c for both data flow and behavioral model using if and else statement with required condition.
iii. Make test bench waveform and check for the given inputs.

Code:

FULL ADDER

```
begin
s<= a xor b xor cin;
cout<= (a and b) or (b and cin) or (a and cin);
end Behavioral;
```

PARALLEL ADDER

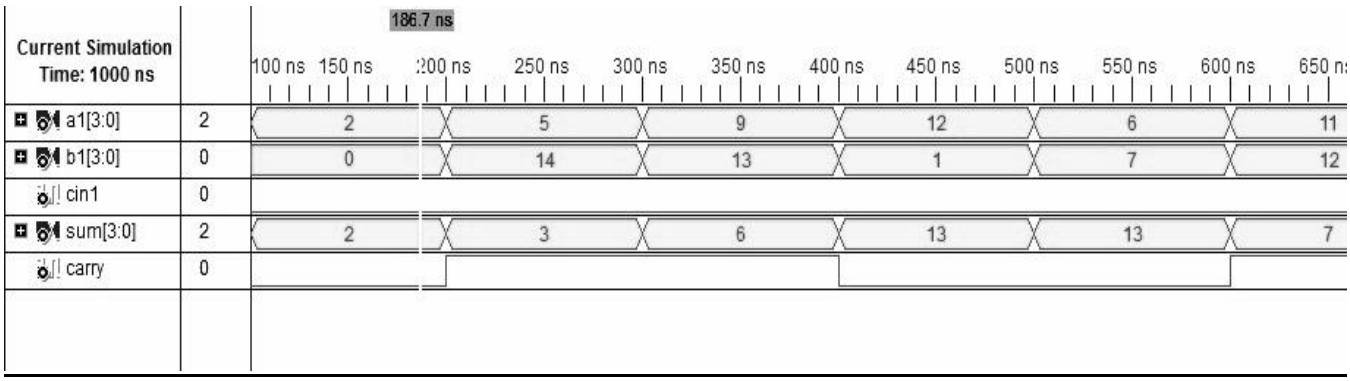
```
entity pa1 is
   Port ( a1 : in  STD_LOGIC_VECTOR (3 downto 0);
          b1 : in  STD_LOGIC_VECTOR (3 downto 0);
          cin1 : in  STD_LOGIC;
          sum : out  STD_LOGIC_VECTOR (3 downto 0);
          carry : out  STD_LOGIC);
end pa1;

architecture Behavioral of pa1 is
          COMPONENT fa
          PORT(
                    a : IN std_logic;
                    b : IN std_logic;
                    cin : IN std_logic;
                    s : OUT std_logic;
                    cout : OUT std_logic
                    );
          END COMPONENT;
```

```
signal x: STD_LOGIC_VECTOR (2 downto 0);
begin
e1: fa port map(a1(0),b1(0),cin1,sum(0),x(0));
e2: fa port map(a1(1),b1(1),x(0),sum(1),x(1));
e3: fa port map(a1(2),b1(2),x(1),sum(2),x(2));
e4: fa port map(a1(3),b1(3),x(2),sum(3),carry);
end Behavioral;
```

Structural Model:

Output:



Name: Ranajit Roy **Roll:** 47 **Sec:** A

Assignment 2: Implementation of 2:1 MUX and 4:1 MUX

Objective: Implement 2:1 Mux using AND, OR, NOT gate

Software used:

Property Name	Value
Device family	Spartan 3
Device	XC3S50
Package	PQ208
Speed	-5
Top-level source type	HDL
Synthesis Tool	XST(VHDL/Verilog)
Simulator	ISE Simulator
Preferred Language	VHDL

Theory:

i. Make VHDL with required port specification.
ii. Calculate the value c for both data flow and behavioral model using if and else statement with required condition.
iii. Make test bench waveform and check for the given inputs.

Structural Model:

Code:

AND GATE

```
begin
c<= a and b;
end Behavioral;
```

OR GATE

```
begin
c<=a or b;
end Behavioral;
```

NOT GATE

```
begin
b<=not a;
end Behavioral;
```

MUX 2:1

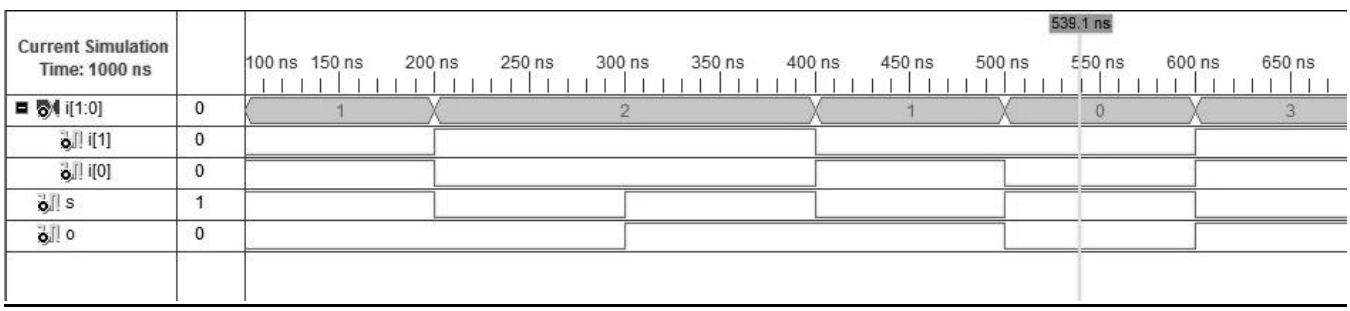
```
entity mux21 is
   Port ( i : in  STD_LOGIC_VECTOR (1 downto 0);
          s : in  STD_LOGIC;
          o : out  STD_LOGIC);
end mux21;

architecture Behavioral of mux21 is

          COMPONENT and1
          PORT(
                    a : IN std_logic;
                    b : IN std_logic;
                    c : OUT std_logic
                    );
          END COMPONENT;
          COMPONENT not1
          PORT(
                    a : IN std_logic;
                    b : OUT std_logic
                    );
          END COMPONENT;
                    COMPONENT or1
          PORT(
                    a : IN std_logic;
                    b : IN std_logic;
                    c : OUT std_logic
                    );
          END COMPONENT;
          signal x,y,z: STD_LOGIC;

begin
a1: and1 port map(s,i(1),x);
a2: not1 port map(s,z);
a3: and1 port map(z,i(0),y);
a4: or1 port map(x,y,o);
end Behavioral;
```

Output:



Name: Ranajit Roy **Roll:** 47 **Sec:** A

Objective: Implement 4:1 MUX using 2:1 MUX

Software used:

Property Name	Value
Device family	Spartan 3
Device	XC3S50
Package	PQ208
Speed	-5
Top-level source type	HDL
Synthesis Tool	XST(VHDL/Verilog)
Simulator	ISE Simulator
Preferred Language	VHDL

Theory:

i. Make VHDL with required port specification.
ii. Calculate the value c for both data flow and behavioral model using if and else statement with required condition.
iii. Make test bench waveform and check for the given inputs.

Structural Model:

Code:

MUX 2:1

```
begin
o<=((not s) and i0) or (s and i1);
end Behavioral;
```

MUX 4:1

```
entity mux41 is
   Port ( i : in  STD_LOGIC_VECTOR (3 downto 0);
          ss : in  STD_LOGIC_VECTOR (1 downto 0);
          outp : out  STD_LOGIC);
end mux41;

architecture Behavioral of mux41 is
          COMPONENT mux21
          PORT(
                    i0 : IN std_logic;
                    i1 : IN std_logic;
                    s : IN std_logic;
                    o : OUT std_logic
                    );
          END COMPONENT;
          signal x,y,z: STD_LOGIC;
begin
a1: mux21 port map(i(0),i(1),ss(0),x);
a2: mux21 port map(i(2),i(3),ss(0),y);
a3: mux21 port map(x,y,ss(1),outp);
end Behavioral;
```

Output: