Index

S. No.	Name of the Experiment	Page No.	Date of Experiment	Date of Submission	Remarks
1.	Design a Half and Full Adder using VHDL				5/11/23
2.	Design Half and Full Subtractor using VHDL				
3.	Design 4 bit adder using macro of half adder and full adder				
4.	Design 4 bit adder subtractor circuit using mode control bit				
5.	Design 3:8 and 2:4 decoder circuit.				
6.	Design 5:32 decoder circuit using macro of 2:4 and 3:8 decoder circuit				
7.	Design 4:1 MUX using when else statement				
8.	Design 2 bit comparator circuit. Design 4 bit comparator circuit using macro of 2 bit comparator circuit.				5/11/23
9.	Design SR, JK, D and T flip flop				

Index

S. No.	Name of the Experiment	Page No.	Date of Experiment	Date of Submission	Remarks
10	Design a serial in serial out 4 bit shift register				
11.	Design universal 4 bit shift register				
12.	Design a 4 bit binary counter				
13.	Design a decade counter				
14.	Design a 4 bit up/down asynchronous counter				
15.	Design a 4 bit multiplier circuit using add shift state machine model				

5/11/23

Date _____

Expt. No. _____ Page No. _____

Aim: Design a half adder and full adder using VHDL

Software Used: Xilinx ISE 14.7

VHDL Code (Half Adder)

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity half_adder is
port (a,b : in bit; s,c : out bit);
end half_adder;
architecture behave of half_adder is
 begin
    s <= a xor b;
    c <= a and b;
 end behave;
```

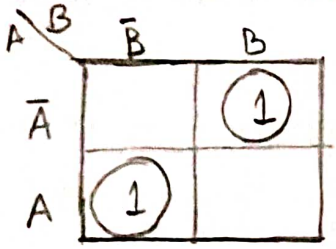
VHDL Code (Full Adder)

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity full_adder is
 port (a, b, ci : in bit; s, c : out bit);
end full_adder;
architecture behave of full_adder is
 signal c1, c2, c3 : bit;
begin
    s <= a xor b xor ci;
    c1 <= a and b;
    c2 <= a xor b;
```

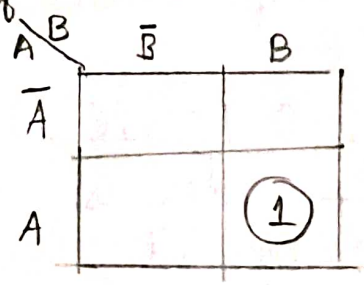
Teacher's Signature : _____

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Truth Table for Half Adder

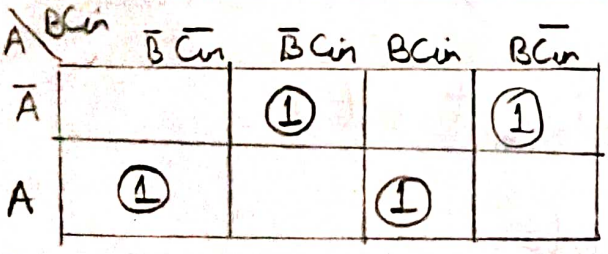


$S = A \oplus B$

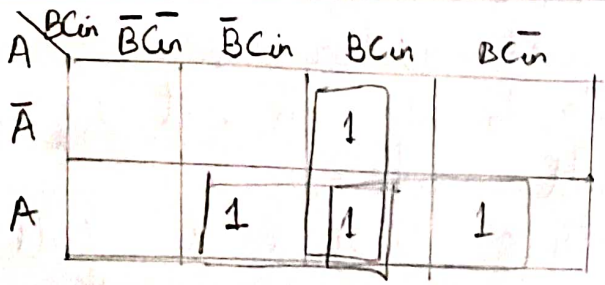


$C = A \cdot B$

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



$S = A \oplus B \oplus C_{in}$

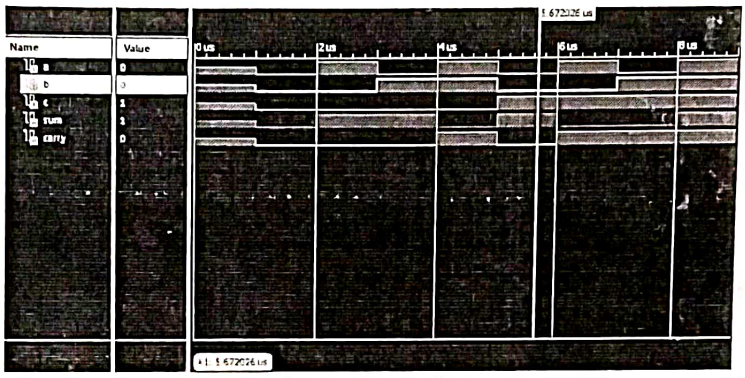


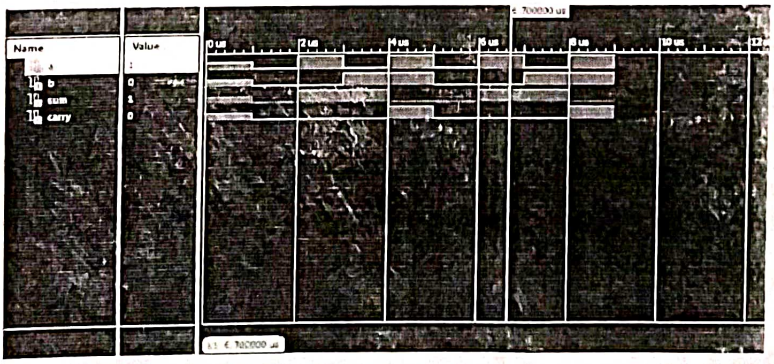
$C_{out} = AB + BC_{in} + C_{in}A$

Date _____

Expt. No. _____ Page No. _____

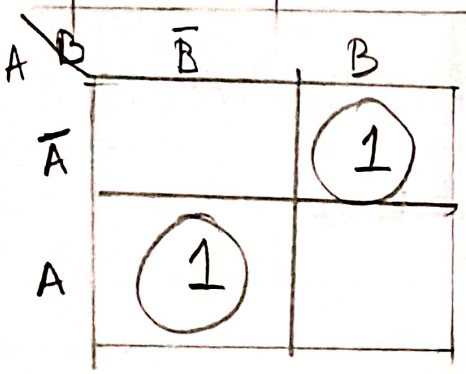
C3 <= Ci and C2;
C <= C3 or C1;
end behave;



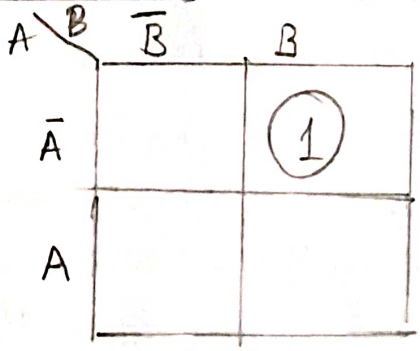


Teacher's Signature : 5/11/23

Inputs		Outputs	
X	Y	D	Bout
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



$S = A \oplus B$



$C = \bar{A} \cdot B$

Inputs			Output	
A	B	Bin	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Date _____

Expt. No. _____ Page No. _____

Aim: Design half and full subtractor using VHDL

Software Used: Xilinx ISE 14.7

VHDL Code (Half Subtractor)

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity half_subt is
 port (a: in STD_LOGIC;
       c: in STD_LOGIC;
       b: out STD_LOGIC;
       d: out STD_LOGIC);
end half_subt;
architecture behavioral of half_subt is
 begin
 process (a,c)
 begin
 if (a='0' and c='0')
 then b<='0'; d<='0';
 elsif (a='0' and c='1')
 then b<='1'; d<='1';
 elsif (a='1' and c='0')
 then b<='0'; d<='1';
  else
 b<='0'; d<='0';
 end if;
 end process;
 end behavioral;
```

Teacher's Signature : _____

A \ B Bin	00	01	11	10
0	0	1	0	1
1	1	0	1	0

$$D = A'B'B_{in} + AB'B_{in}' + A'BB_{in}' + ABB_{in}$$

A \ B.Bin	00	01	11	10
0	0	1	1	1
1	0	0	1	0

$$B_{out} = A'B_{in} + A'B + BB_{in}$$

Date _____

Expt. No. _____ Page No. _____

VHDL Code (Full Subtractor)

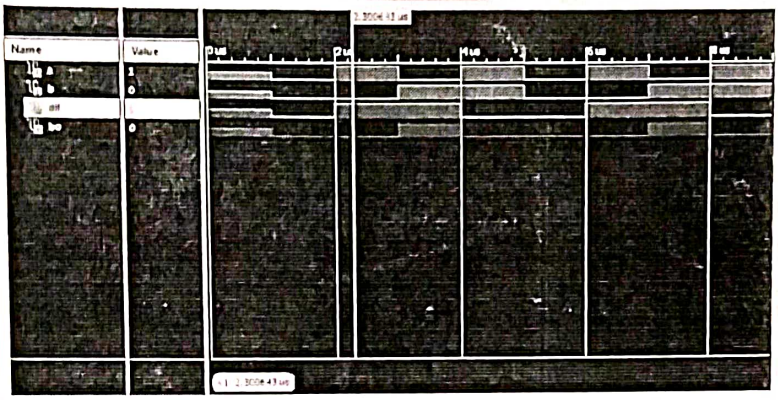
```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity full_subt is
  port (a : in STD_LOGIC;
        b : in STD_LOGIC;
        c : in STD_LOGIC;
     bout : OUT STD_LOGIC;
        d : OUT STD_LOGIC);
end full_subt;
architecture behavioral of full_subt is
begin
process (a, b, c)
begin
if (a='0' and b='0' and c='0')
then bout <='0'; d<='0';
elsif (a='0' and b='0' and c='1')
then bout <='1'; d<='1';
elsif (a='0' and b='1' and c='0')
then bout <='1'; d<='1';
elsif (a='0' and b='1' and c='1')
then bout <='1'; d<='0';
elsif (a='1' and b='0' and c='1')
then bout <='0'; d<='0';
elsif (a='1' and b='1' and c='0')
then bout <='0'; d<='0';
else bout <='1'; d<='1';
end if;
```

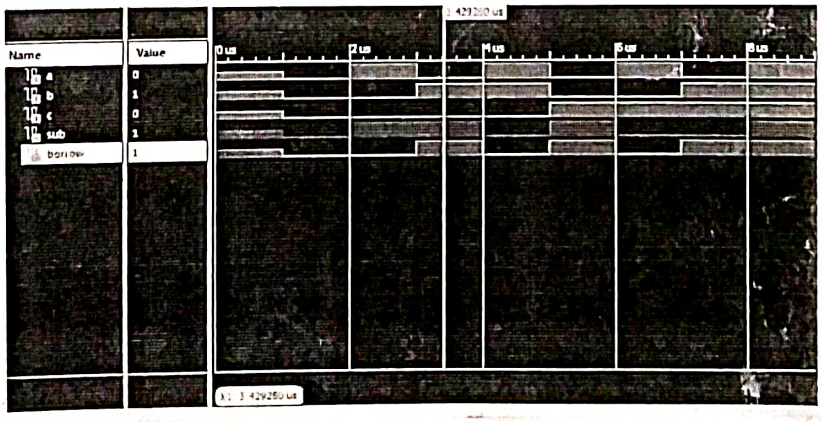
Teacher's Signature : _____

Date _____

Expt. No. _____ Page No. _____

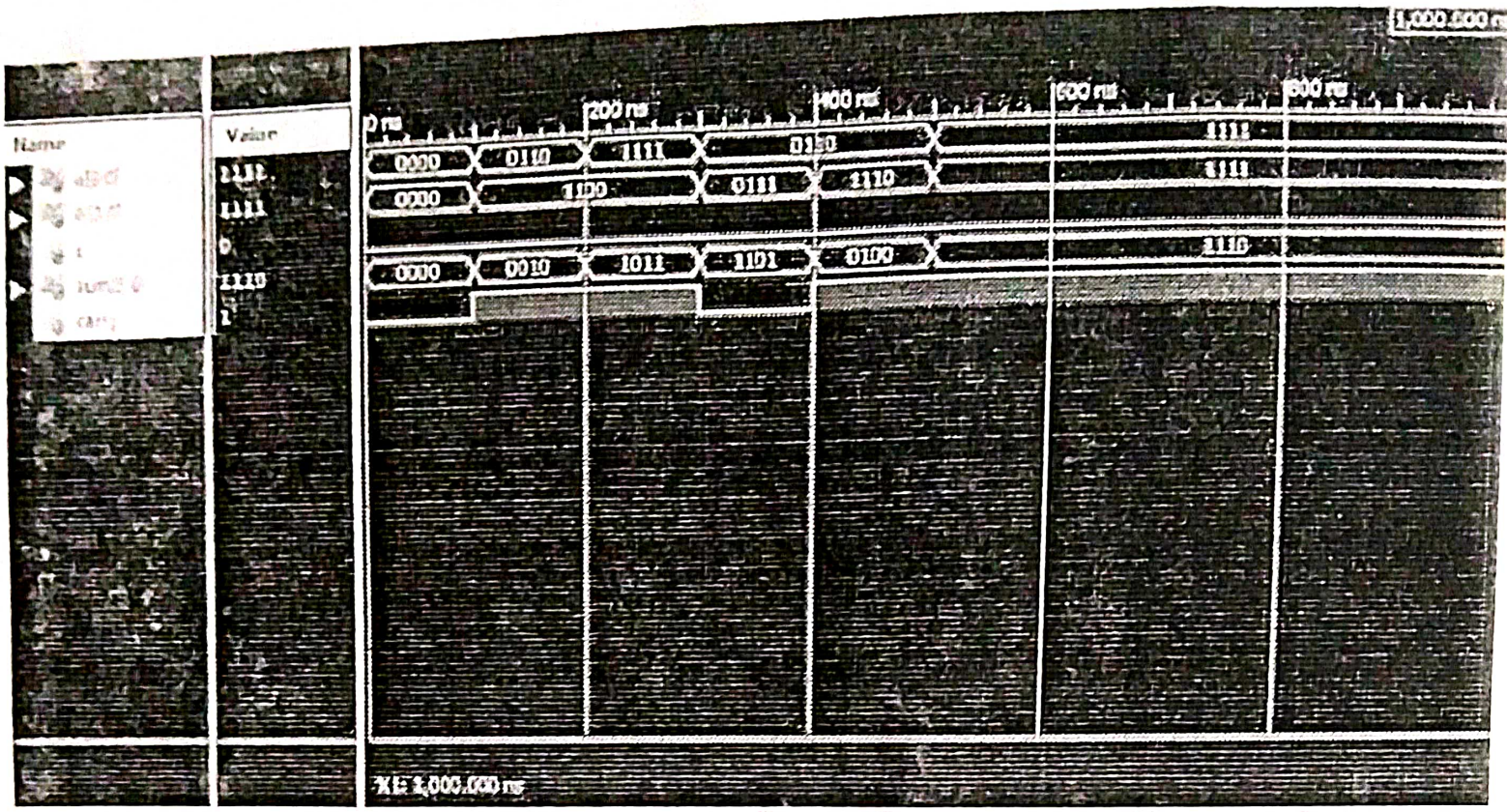
```
end process;
end behavioral;
```





5/11/23

Teacher's Signature : _____


Name
Value
200 ns
400 ns
600 ns
800 ns
0000
0110
1111
0000
1100
0111
1110
0000
0010
1011
1101
0100
1110
X1: 1,000.000 ns

Date _____

Expt. No. _____ Page No. _____

Aim: Design a 4 bit adder using macro of half adder and full adder

Software Used: Xilinx ISE 14.7

VHDL Code:

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity hadder is
port (c, a, b : in STD_LOGIC_VECTOR (3 downto 0);
      c: in STD_LOGIC;
      s: out STD_LOGIC_VECTOR (3 downto 0);
      CO : out STD_LOGIC);
end padder;
architecture structural of padder is
component fadder
port ( a, b, c : in STD_LOGIC;
       su, cy : out STD_LOGIC);
end component;
signal x, y, z : STD_LOGIC;
begin
fa1 : fadder port map (c, a(0), b(0), s(0), x);
fa2 : fadder port map (x, a(1), b(1), s(1), y);
fa3 : fadder port map (y, a(2), b(2), s(2), z);
fa4 : fadder port map (z, a(3), b(3), s(3), cO);
end structural;
```

Teacher's Signature : _____ 11123

Date _____

Expt. No. _____ Page No. _____

Aim: Design 4 bit adder using macro of half adder and full adder.

Software Used: Xilinx 14.4 ISE

VHDL Code:

```
library IEEE;
use IEEE.STD_LOGIC_ARITH.all;
use IEEE.STD_LOGIC_1164.all;
use IEEE.STD_LOGIC_UNSIGNED.all;
entity ha is
  port (x,y : in STD_LOGIC;
        s,c : out STD_LOGIC;);
  end ha;
architecture behavioral of ha is
begin
    s <= x nor y;
    c <= x and y;
end behavioral;
entity fa is
port (a,b,cin : in STD_LOGIC;
   sum,carry : out STD_LOGIC;);
  end fa;
architecture behavioral of fa is
   component ha is
port (x,y : in STD_LOGIC;
      s,c : out STD_LOGIC;);
  end component;
```

Teacher's Signature : _____

Date _____

Expt. No. _____ Page No. _____

```
signal temp1, temp2, temp3 : STD_LOGIC;
begin
  ha1 : ha port map ( x => a, y => b, s => temp1, c => temp2);
  ha2 : ha port map (x => temp1, y => cin, s => sum, c => temp3);
  carry <= temp2 or temp3;
end behavioral;
entity adder4bit is
  port ( a : in STD_LOGIC_VECTOR ( 3 downto 0);
         b : in STD_LOGIC_VECTOR (3 downto 0);
  carry in : in STD_LOGIC;
       sum : out STD_LOGIC_VECTOR (3 downto 0);
  carry out : out STD_LOGIC;
end adder4bit;
architecture behavioral of adder4bit is
  component fa is
port ( a, b, cin : in STD_LOGIC;
       sum, carry : out STD_LOGIC; );
  end component fa;
signal temp : STD_LOGIC_VECTOR (2 downto 0);
  begin
  u1 : fa port map (a => a(0), b => b(0), cin => carryin,
       sum => sum(0), carry => temp(0));
  u2 : fa port map (a => a(1), b => b(1), cin => temp(0),
       sum => sum(1), carry => temp(1));
  u3 : fa port map (a => a(2), b => b(2), cin => temp(1),
       sum => sum(2), carry => temp(2));
  u4 : fa port map (a => a(3), b => b(3), cin => temp(2),
       sum => sum(3), carry => carryout);
```

Teacher's Signature : _____

Date _____

Expt. No. _____ Page No. _____

```
end behavioral;
```

Teacher's Signature : _____

M	B	X
0	0	0
0	1	1
1	0	1
1	1	0

If M=0 → output is same as B input
If M=1 → output is complement of B input

For Sum

A \ BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
$\bar{A}$	0	1	0	1
A	1	0	1	0

For Carry

A \ BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
$\bar{A}$	0	0	1	0
A	0	1	1	1

For Borrow

A \ BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
$\bar{A}$	0	1	1	1
A	0	0	1	0

Boolean Expression

Sum / Subtract : A XOR B XOR C
Carry : $AB + BC + CA$
Borrow : $\bar{A}C + \bar{A}B + BC$

Date _____

Expt. No. _____ Page No. _____

Aim: Design 4 bit adder subtractor circuit using mode control bit.

Software used: Xilinx 14.4 ISE

Code:

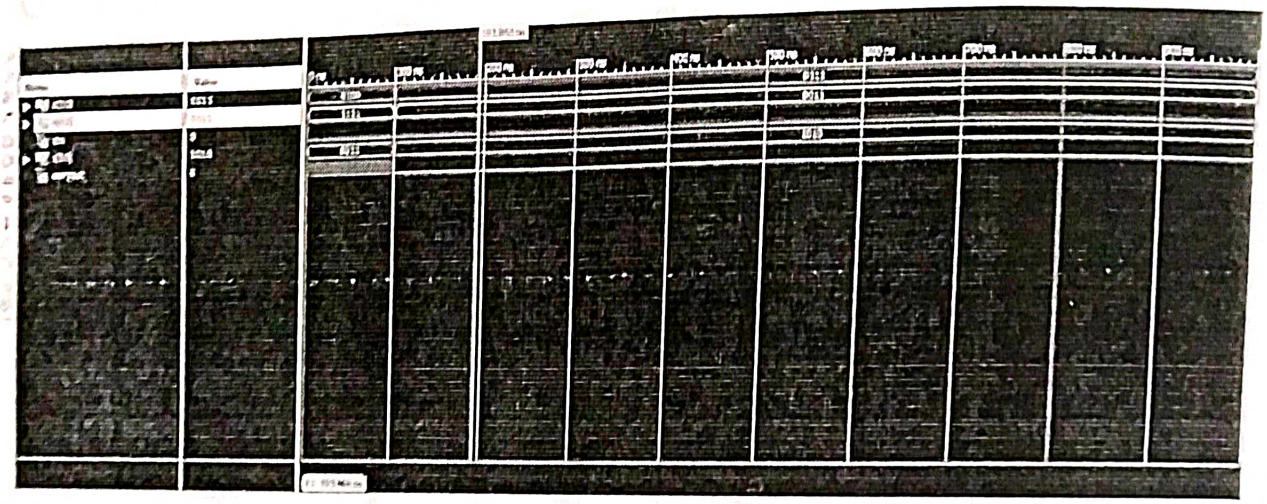
Full Adder

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity Full_Adder is
  port( x,y, Cin : in STD_LOGIC;
        sum, count : out STD_LOGIC);
end Full_Adder;
architecture bhv of Full_Adder is
begin
  sum <= (x xor y) xor cin;
  cout <= (x and (y or cin)) or (cin and y);
end bhv;
```

4 Bit Adder Subtractor

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity addsub is
  port( op : in STD_LOGIC;
        a,b : in STD_LOGIC_VECTOR(3 downto 0);
        r : out STD_LOGIC_VECTOR(3 downto 0);
        cout, overflow : out STD_LOGIC);
end addsub;
```

Teacher's Signature : _____



Date _____

Expt. No. _____ Page No. _____

```
architecture struct of addsub is
 component Full_Adder is
 port( x, y, cin : in STD_LOGIC;
    sum, cout : out STD_LOGIC);
 end component;
 signal c1, c2, c3, c4 : STD_LOGIC;
 signal tmp : STD_LOGIC_VECTOR(3 downto 0);
 begin
    tmp <= a xor b;
FA0 : Full_Adder port map (A(0), tmp(0), op, r(0), c1);- r0
FA1 : Full_Adder port map(A(1), tmp(1), c1, r(1), c2);- r1
FA2 : Full_Adder port map(A(2), tmp(2), c2, r(2), c3);- r2
FA3 : Full_Adder port map (A(3), tmp(3), c3, r(3), c4);- r3
 overflow <= c3 xor c4;
  cout <= c4;
  end struct;
```

Teacher's Signature : _____ 5/11/31

D(1)	D(0)	Y(3)	Y(2)	Y(1)	Y(0)
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

$Y(0) = \overline{D(1)} \cdot \overline{D(0)}$

$Y(1) = \overline{D(1)} \cdot D(0)$

$Y(2) = D(1) \cdot \overline{D(0)}$

$Y(3) = D(1) \cdot D(0)$

A	B	C	D0	D1	D2	D3	D4	D5	D6	D7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

$D_0 = \bar{A}\bar{B}\bar{C}$

$D_1 = \bar{A}\bar{B}C$

$D_2 = \bar{A}B\bar{C}$

$D_3 = \bar{A}BC$

$D_4 = A\bar{B}\bar{C}$

$D_5 = A\bar{B}C$

$D_6 = AB\bar{C}$

$D_7 = ABC$

Date _____

Expt. No. _____ Page No. _____

Aim: Design 3:8 and 2:4 decoder circuit

Software Used: Xilinx 14.4 ISE

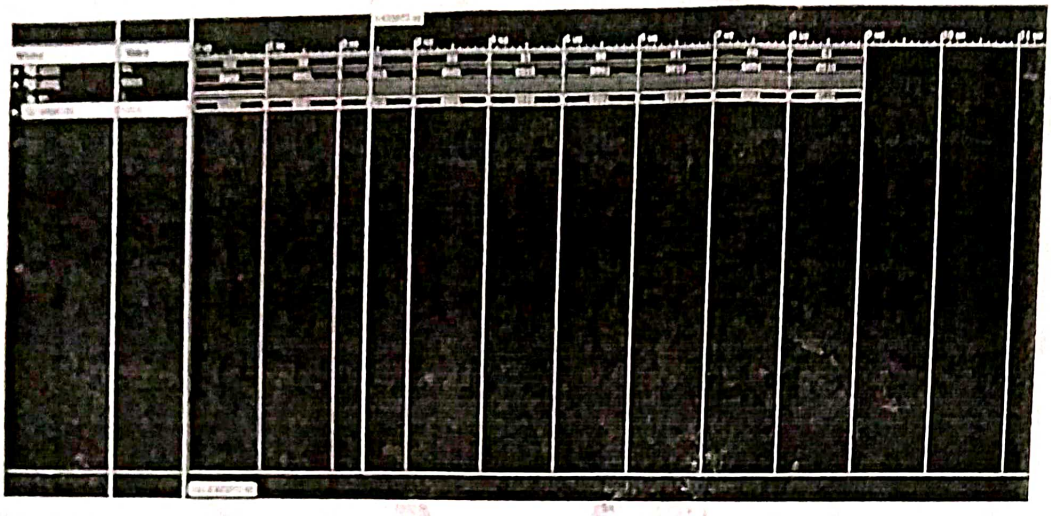
VHDL Code (2:4 Decoder)

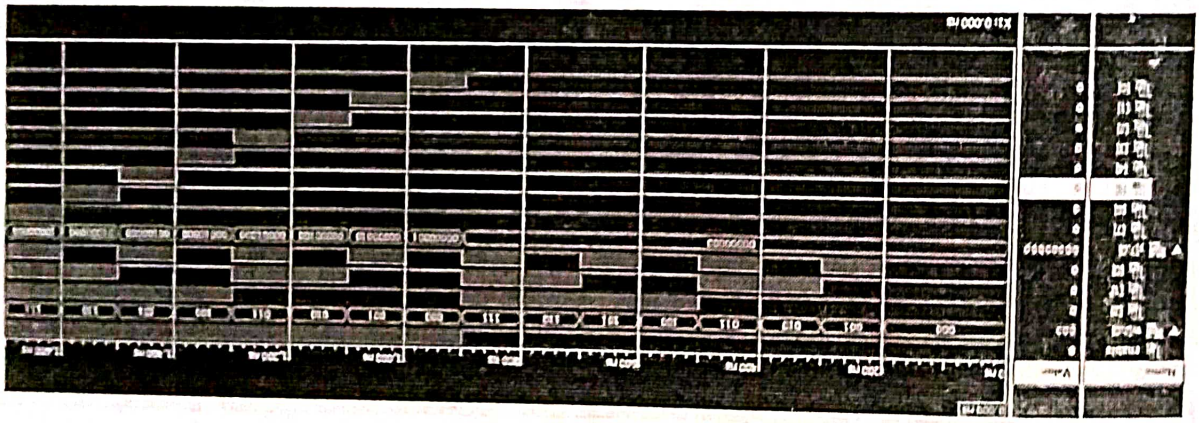
```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity decoder24 is
port (x: in STD_LOGIC_VECTOR(1 downto 0);
      y: out STD_LOGIC_VECTOR(3 downto 0);
      en: in STD_LOGIC;);
end decoder24;
architecture behv of decoder2_4 is
signal temp: STD_LOGIC_VECTOR(3 downto 0);
  begin
temp <= "0001" when x="00" else
  "0010" when x="01" else
  "0100" when x="10" else
  "1000" when x="11";
y <= temp when en='1' else "0000";
end behav;
```

VHDL code (3:8 decoder)

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity decoder3_8 is
port(w: in STD_LOGIC_VECTOR(2 downto 0);
     u: out STD_LOGIC_VECTOR(7 downto 0);
```

Teacher's Signature : _____

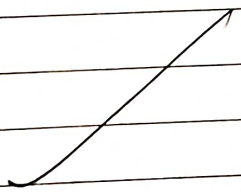




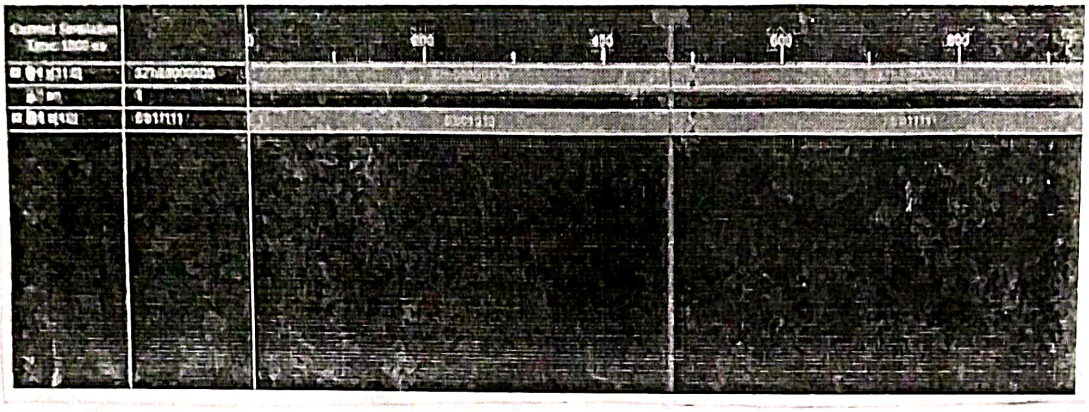
Date _____

Expt. No. _____ Page No. _____

```
en-3 : in STD_LOGIC);
end decoder3-8;
architecture beh of decoder3-8 is
signal temp3 : STD-LOGIC_VECTOR(7 downto 0);
begin
temp3 <= "00000001" when w = "000" else
"00000010" when w = "001" else
"00000100" when w = "010" else
"00001000" when w = "011" else
"00010000" when w = "100" else
"00100000" when w = "101" else
"01000000" when w = "110" else
"10000000";
u <= temp3 when en3 = '1' else "00000000";
end beh;
```



Teacher's Signature : _____



Date _____

Expt. No. _____ Page No. _____

Aim: Design a 5:32 decoder circuit using macro of 2:4 and 3:8 decoder circuit.

Software Used: Xilinx 14.4 ISE

VHDL Code:

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity decoder5_32 is
port( m: in STD_LOGIC_VECTOR(4 downto 0);
  n: out STD_LOGIC_VECTOR(31 downto 0);
  en5: in STD_LOGIC);
    end decoder5_32;
architecture behavioral of decoder5_32 is
  component decoder2_4 is
port(x: in STD_LOGIC_VECTOR(1 downto 0);
 y: out STD_LOGIC_VECTOR(3 downto 0);
 en: in STD_LOGIC);
end component decoder2_4;
component decoder3_8 is
  port(w: in STD_LOGIC_VECTOR(2 downto 0);
    u: out STD_LOGIC_VECTOR(7 downto 0);
    en3: in STD_LOGIC);
  end component decoder3_8;
signal temp5: STD_LOGIC_VECTOR(3 downto 0);
 begin
a2_4: Decoder2_4 port map(x(1) => m(4), x(0) => m(3),
    y(3) => temp5(3), y(2) => temp5(2), y(1) => temp5(1),
```

Teacher's Signature : _____

Date _____

Expt. No. _____ Page No. _____

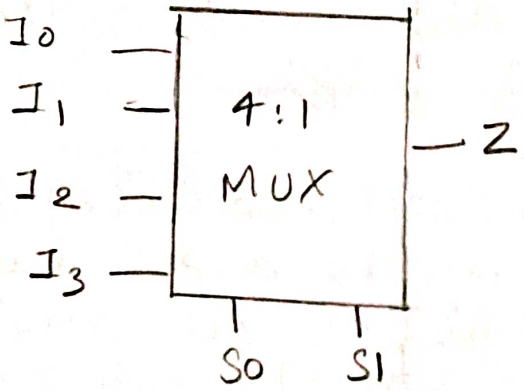
y(0) => temp5(0), en => en5);
a3_8_0: decoder3_8 port map (w(2) => m(2), w(1) => m(1), w(0) => m(0), en_3 => temp5(0), u(7) => n(7), u(6) => n(6), u(5) => n(5), u(4) => n(4), u(3) => n(3), u(2) => n(2), u(1) => n(1), u(0) => n(0));
a3_8_1: decoder3_8 port map (w(2) => m(2), w(1) => m(1), w(0) => m(0), en_3 => temp5(1), u(7) => n(15), u(6) => n(14), u(5) => n(13), u(4) => n(12), u(3) => n(11), u(2) => n(10), u(1) => n(9), u(0) => n(8));

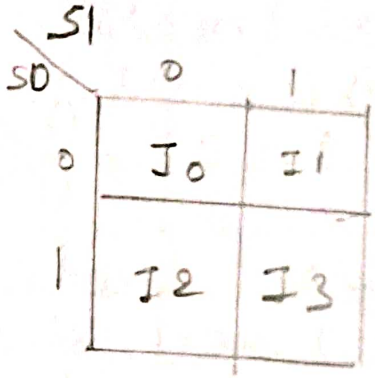
a3_8_2: decoder3_8 port map (w(2) => m(2), w(1) => m(1), w(0) => m(0), en_3 => temp5(2), u(7) => n(23), u(6) => n(22), u(5) => n(21), u(4) => n(20), u(3) => n(19), u(2) => n(18), u(1) => n(17), u(0) => n(16));

a3_8_3: decoder3_8 port map (w(2) => m(2), w(1) => m(1), w(0) => m(0), en_3 => temp5(3), u(7) => n(31), u(6) => n(30), u(5) => n(29), u(4) => n(28), u(3) => n(27), u(2) => n(26), u(1) => n(25), u(0) => n(24));
end behavioral;

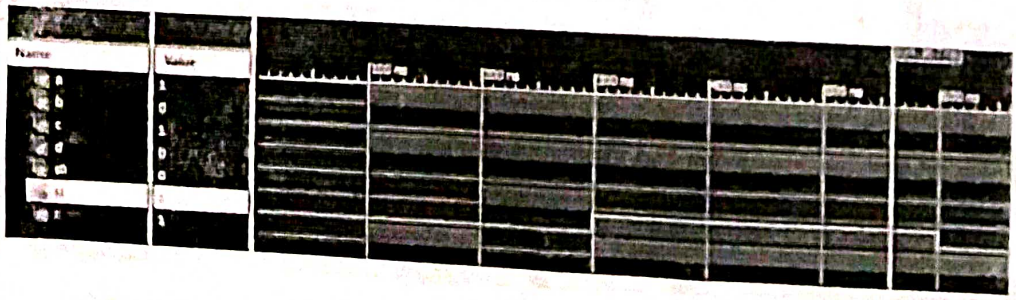
Teacher's Signature : _____

S(0)	S(1)	Z
0	0	a(0)
0	1	a(1)
1	0	a(2)
1	1	a(3)





$$Z = \overline{S0}\,\overline{S1}\,I0 + \overline{S0}\,S1\,I1 + S0\,\overline{S1}\,I2 + S0\,S1\,I3$$



Date

Expt. No. Page No.

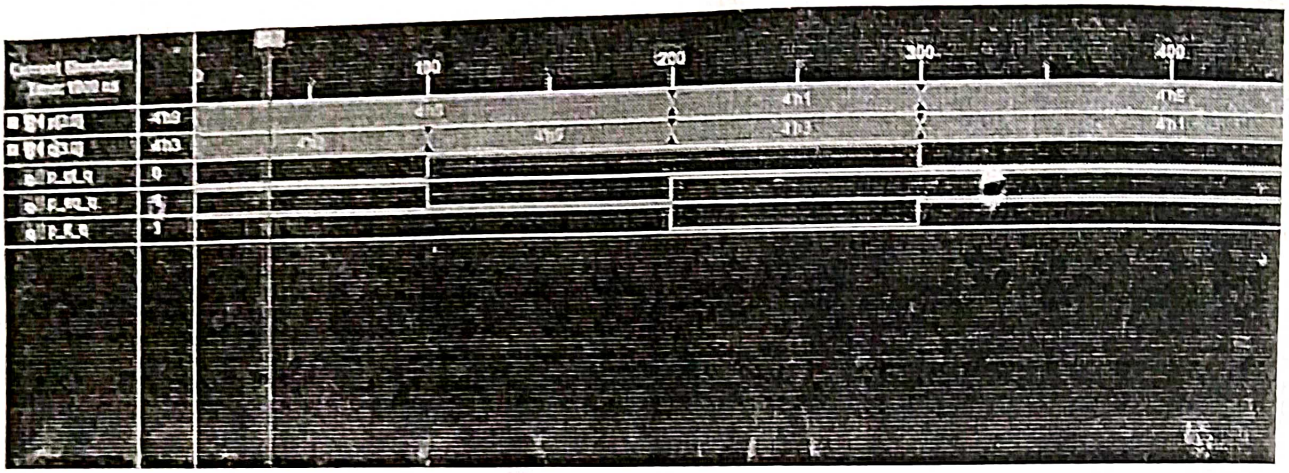
Aim : Design 4:1 MUX using when else statement.

Software Used : Xilinx 14.4 ISE

VHDL Code

```
Library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity mux4x1selsat is
port (a: in STD_LOGIC_VECTOR(3 downto 0);
      s: in STD_LOGIC_VECTOR(1 downto 0);
      z: out STD_LOGIC);
end mux4x1selsat;
architecture bhv1 of mux4x1selsat is
begin
z <= a(0) when s = "00" else
     a(1) when s = "01" else
     a(2) when s = "10" else
     a(3) when s = "11" ;
end bhv1;
```

Teacher's Signature : 5/11/23


100
200
300
400

Date _____

Expt. No. _____ Page No. _____

Aim: Design 2 bit comparator circuit. Design 4 bit comparator circuit using macro of 2 bit comparator circuit.

Software Used: Xilinx 14.4 ISE

Code (2 Bit Comparator)

```
library ieee;
use ieee.std_logic_1164.all;
entity comp2 is
port (
  a, b : in std_logic_vector (1 downto 0);
  it, st, eq : out std_logic
  );
  end comp2;
architecture bit2_comparator of comp2 is
signal p0, p1, p2, p3, p4, p5, p6, p7, p8, p9 : std_logic;
begin
  eq <= p0 or p1 or p2 or p3;
p0 <= ((not a(1)) and (not a(0)) and (not b(1))
    and (not b(0)));
p1 <= ((not a(1)) and a(0) and (not b(1)) and b(0));
p2 <= (a(1) and (not a(0)) and b(1) and (not b(0)));
p3 <= (a(1) and a(0) and b(1) and b(0));
  st <= p9 or p5 or p6;
p4 <= ((not a(0)) and b(1) and b(0));
p5 <= ((not a(1)) and (not a(0)) and b(0));
p6 <= ((not a(1)) and b(1));
```

Teacher's Signature : _____ 5/11/23

Date _____

Expt. No. _____ Page No. _____

```
it <= p7 or p8 or p9;
p7 <= ( a(1) and a(0) and (not b(0));
p8 <= ( a(1) and (not b(1)) );
p9 <= (a(0) and ( not b(1)) and (not b(0)));
end bit2 - comparator;
```

4 Bit Comparator

```
library ieee;
use ieee.std_logic_1164.all;
entity comp4 is
port(
   a,b : in std_logic_vector (3 downto 0);
   it, st, eq : out std_logic;
LED : out std_logic_vector (6 downto 0)
   );
end comp4;
architecture bit4_comparator of comp4 is
signal e1, e0, s1, s0, i1, i0 : std_logic;
begin
comp1_unit : entity work.comp2 (bit2_comparator)
port map (a(1) => a(3), a(0) => a(2), b(1) => b(3),
      b(0) => b(2), eq => e1, st => s1, it => i1);
comp2_unit : entity work.comp2 (bit2_comparator)
port map (a(1) => a(1), a(0) => a(0), b(1) => b(1), b(0) => b(0),
   eq => e0, st => s0, it => i0);
   eq <= e1 and e0;
   it <= i1 or ( e1 and i0);
   st <= s1 or ( e1 and s0);
end bit4_comparator;
```

Teacher's Signature : _____ 5/11/23

θ	S	R	$\theta(T+1)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	—
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	—

θ	J	K	$\theta(T+1)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

θ	D	$\theta(T+1)$
0	0	0
0	1	1
1	0	0
1	1	1

θ	T	$\theta(T+1)$
0	0	0
0	1	1
1	0	1
1	1	0

Date _____

Expt. No. _____ Page No. _____

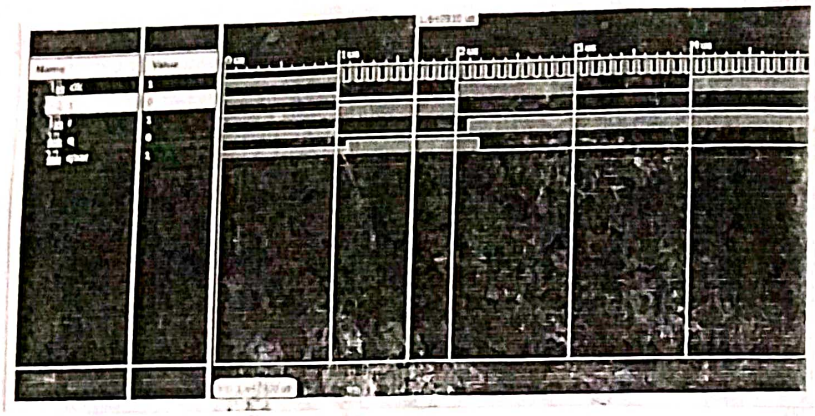
Aim: Design SR, JK, D and T flip flop

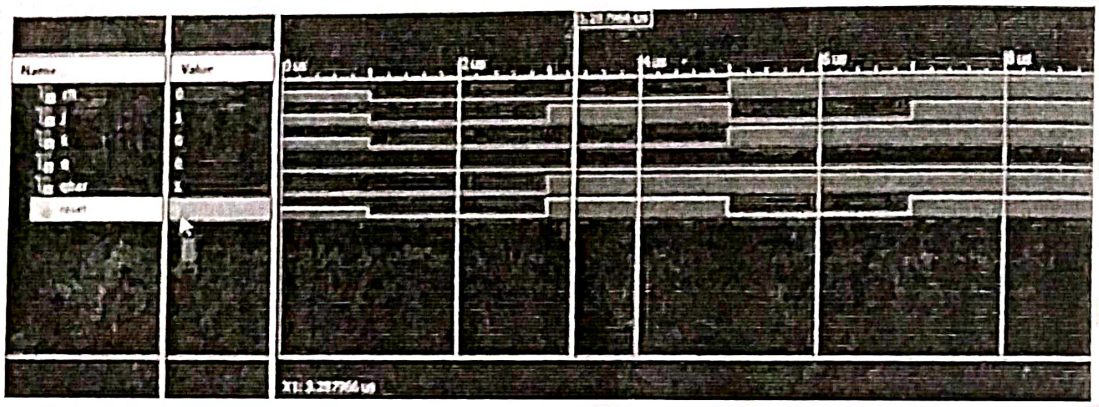
Software Used: Xilinx 14.4 ISE

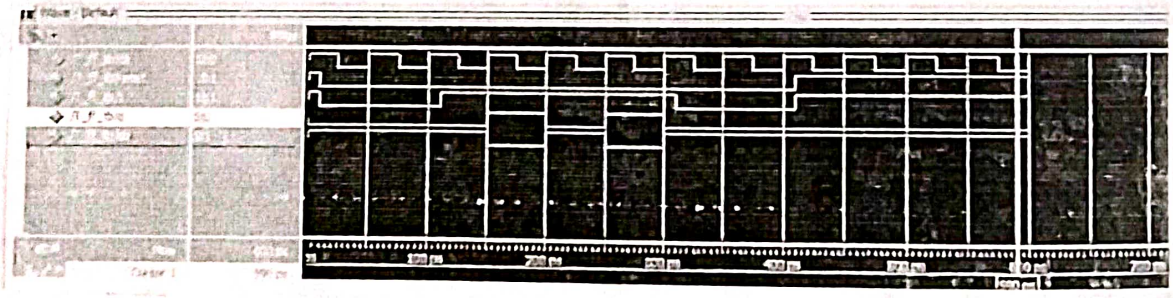
SR Flip Flop

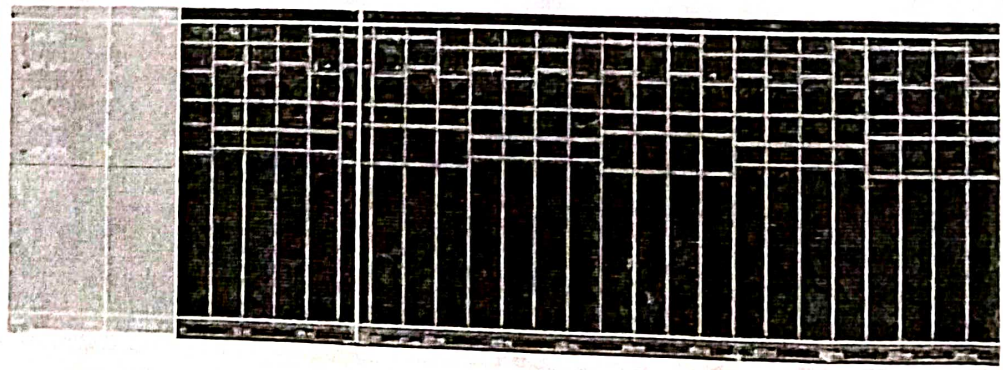
```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity sr_flipflop_source is
port (s, r, rst, clk : in std_logic;
      q, qb : out std_logic);
end sr_flipflop_source;
architecture behavioral of sr_flip_flop_source is
begin
process (s, r, rst, clk)
begin
if (rst = '1') then
  q <= '0';
elsif (rising_edge (clk)) then
  if (s /= r) then
    q <= s;
    qb <= r;
elsif (s = '1' and r = '1') then
    q <= 'Z';
    qb <= 'Z';
  end if;
  end if;
end process;
end behavioral;
```

Teacher's Signature : _____ 5/11/23









Date _____

Expt. No. _____ Page No. _____

JK Flip Flop

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity JK_FF is
 port( J, K, clk, rst : in std_logic;
       q, qbar : out std_logic);
end JK_FF;
architecture behavioral of JK_FF is
begin
process (clk, rst)
variable qn : std_logic;
begin
if (rst = '1') then
 qn := '0';
elsif (clk'event and clk = '1') then
 if (J = '0' and K = '0') then
  qn := qn;
elsif (J = '0' and K = '1') then
  qn := '0';
elsif (J = '1' and K = '0') then
  qn := '1';
elsif (J = '1' and K = '1') then
  qn := not qn;
else
null;
end if;
```

Teacher's Signature : _____ 5/11/23

```
q <= qn;
qbar <= not qn;
end process;
end behavioral;
```

D Flip Flop

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity d_flipflop_source is
port( D, CLK, RST : in std_logic;
  q, qb : out std_logic);
end d_flipflop_source;
architecture behavioral of d_flipflop_source is
begin
process( D, CLK, RST)
 begin
 if ( RST = '1' ) then
  q <= '0';
elsif (rising_edge (clk)) then
 q <= D;
 qb <= not D;
end if;
end process;
 end behavioral;
```

Date _____

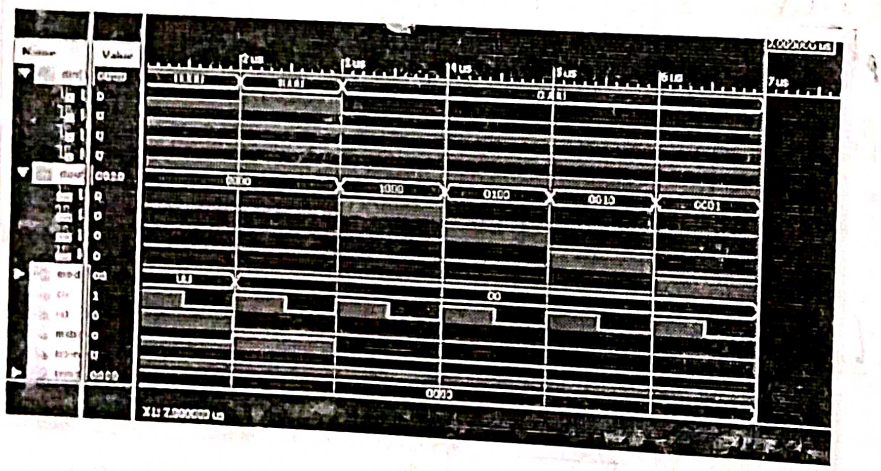
Expt. No. _____ Page No. _____

T Flip Flop

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity T_FLIPFLOP_SOURCE is
port( T, CLK, RST, TEMP : in std_logic;
      Q, QB : out std_logic);
end T_FLIPFLOP_SOURCE;
architecture behavioral of T_FLIPFLOP_SOURCE is
begin
process (T, CLK, RES)
variable TEMP : std_logic := '0';
begin
if (RES = '1') then
TEMP := '0';
elsif (rising_edge (CLK)) then
if (T = '1') then
TEMP := not TEMP;
QB <= TEMP;
end process;
end behavioral;
```

Teacher's Signature : _____ 5/11/23

clk	θ_3	θ_2	θ_1	θ_0
Reset	0	0	0	0
1st falling edge	1	0	0	0
2nd falling edge	1	1	0	0
3rd falling edge	1	1	1	0
4th falling edge	1	1	1	1



Date _____

Expt. No. _____ Page No. _____

Aim: Design a 4 bit Serial In Serial Out shift register

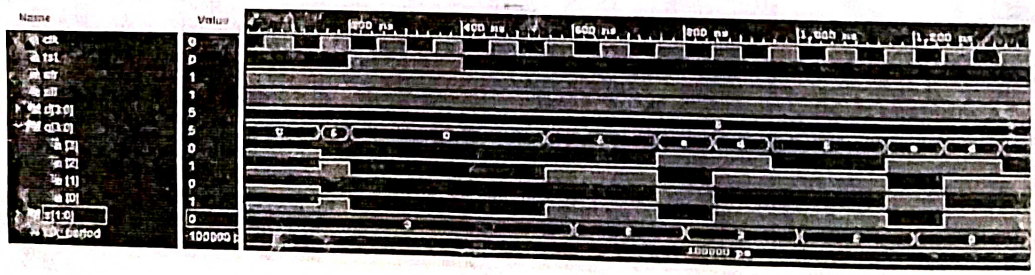
Software Used: Xilinx 14.4 ISE

VHDL Code

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity siso_behaviour is
port(
    din : in STD_LOGIC;
    clk : in STD_LOGIC;
  reset : in STD_LOGIC;
   dout : out STD_LOGIC;
);
end siso_behaviour;
architecture siso_behaviour_arc of siso_behavior is
begin
  siso : process ( clk, din, reset) is
variable s: std_logic_vector( 3 downto 0) := "0000";
begin
  if ( reset = '1' ) then
    s := "0000";
elsif (rising_edge (clk)) then
    s := (din & s( 3 downto 1 ));
      dout <= s(0);
  end if;
end process siso;
end siso_behaviour_arc;
```

Teacher's Signature : _____ 511123

R_CLR	Mode S1 S0	C4 CLK	Serial in SL SR	Parallel in A B C D	Outputs
L	XX	X	XX	XXXX	LLLL
H	XX	$\hat{A}'+\hat{A}$	XX	XXXX	$\varphi_A\varphi_B\varphi_C\varphi_D$
H	HH	$\hat{A}$	XX	a b c d	a b c d
H	LH	$\hat{A}$	X H	XXXX	H $\varphi_A\varphi_B\varphi_C$
H	LH	$\hat{A}$	XL	XXXX	L $\varphi_A\varphi_B\varphi_C$
H	HL	$\hat{A}$	HX	XXXX	$\varphi_B\varphi_C\varphi_D$ H
H	HL	$\hat{A}$	LX	XXXX	$\varphi_B\varphi_C\varphi_D$ L
H	LL	X	XX	XXXX	$\varphi_A\varphi_B\varphi_C\varphi_D$



Date _____

Expt. No. _____ Page No. _____

Aim: Design universal 4 bit shift register

Software Used: Xilinx 14.4 ISE

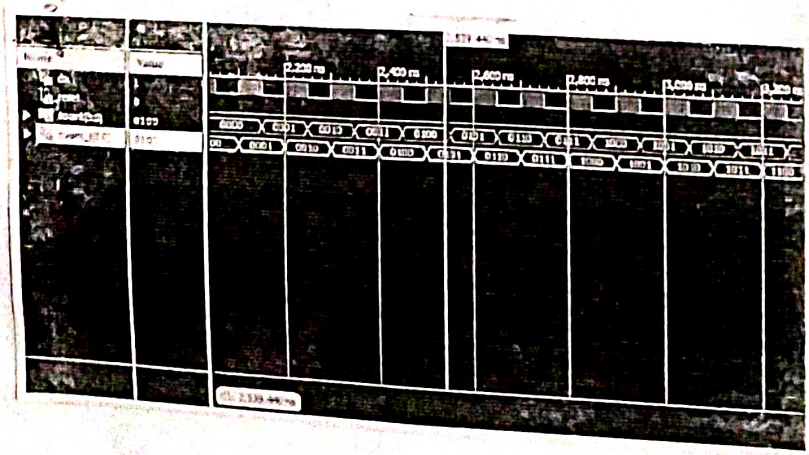
VHDL Code:

```
Library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity uni_shift is
port (clock, clear, sl_in, sr_in : in bit;
mode : in bit_vector (1 downto 0);
data : in bit_vector (3 downto 0);
q : inout bit_vector (3 downto 0));
end uni_shift;
architecture behav of uni_shift is
begin
process (clock, clear)
begin
if clear = '0' then
q <= "0000";
elsif clock'event and clock = '1' then
case model is
when "00" => null;
when "01" => q <= (q srl 1) or (sr_in & "000");
when "10" => q <= (q sll 1) or ("000" & sl_in);
when "11" => q <= data;
end case;
end if;
end process;
end behav;
```

5/11/23

Teacher's Signature : _____

Reset	clk	Q_3	Q_2	Q_1	Q_0
1	0	0	0	0	0
0	1	0	0	0	1
0	2	0	0	1	0
0	3	0	0	1	1
0	4	0	1	0	0
0	5	0	1	0	1
0	6	0	1	1	0
0	7	0	1	1	1
0	8	1	0	0	0
0	9	1	0	0	1
0	10	1	0	1	0
0	11	1	0	1	1
0	12	1	1	0	0
0	13	1	1	0	1
0	14	1	1	1	0
0	15	1	1	1	1
0	16	0	0	0	0



Date _____

Expt. No. _____ Page No. _____

Aim: Design a 4 bit binary counter

Software Used: Xilinx 14.4 ISE

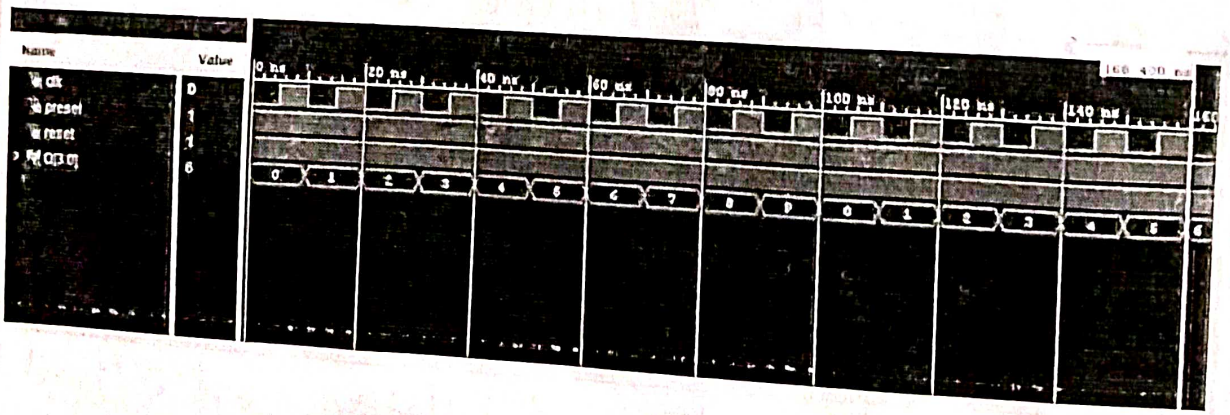
VHDL Code:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.all;
use IEEE.STD_LOGIC_UNSIGNED.all;
entity Counter is
port (rst, clk : in std_logic;
      o : out std_logic_vector (0 to 3));
end counter;
architecture count_arch of counter is
signal count : std_logic_vector (0 to 3);
begin
 process (rst, clk)
  begin
if (rst = '1') then count <= "0000";
elsif (clk'event and clk = '1') then count <= count+1;
end if;
  end process;
  o <= count;
end count_arch;
```

5/11/23

Teacher's Signature : _____

Clock Pulse	X_3	X_2	X_1	X_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10(reset)	0	0	0	0



Date _____

Expt. No. _____ Page No. _____

Aim: Design a decade counter

Software Used: Xilinx 14.4 ISE

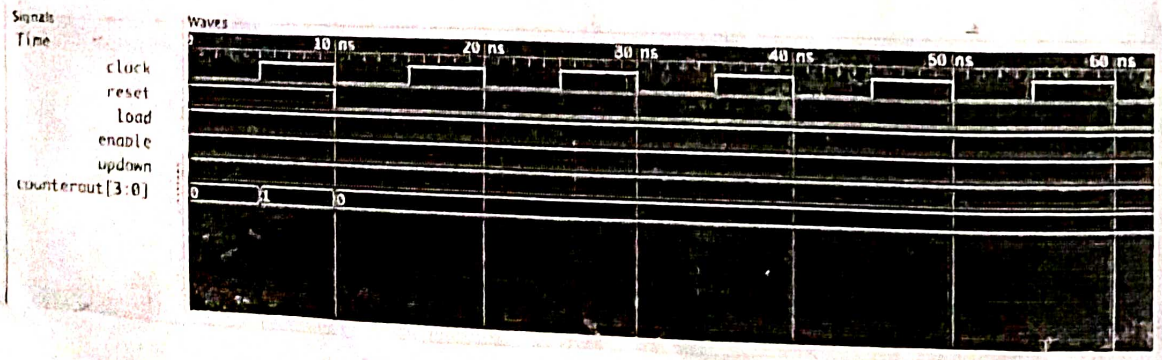
VHDL Code:

```
library IEEE;
use IEEE. STD_LOGIC_1164. ALL;
use IEEE. STD_LOGIC_ARITH. ALL;
use IEEE. STD_LOGIC_UNSIGNED. ALL;
entity mod a is
architecture behave of mod a is
signal temp : STD_LOGIC_VECTOR (3 downto 0);
begin
 a. process (clk)
begin
if ( clk'event and clk ='0' ) then
if ( temp = "1001" ) then temp <= "0000";
else
 temp <= temp + "0001";
end if;
end if;
z <= temp;
end process a;
end behavioral;
```

5/11/23

Teacher's Signature : _____

CLK	Updown	θ_3	θ_2	θ_1	θ_0
0	1	1	1	1	1
1	1	1	1	1	0
2	1	1	1	0	1
3	1	1	1	0	0
4	1	1	0	1	1
5	1	1	0	1	0
6	1	1	0	0	1
7	1	1	0	0	0
8	1	0	1	1	1
9	1	0	1	1	0
10	1	0	1	0	1
11	1	0	1	0	0
12	1	0	0	1	1
13	1	0	0	1	0
14	1	0	0	0	1
15	1	0	0	0	0



Date _____

Expt. No. _____ Page No. _____

Aim: Design a 4 bit up/down asynchronous counter

Software Used: Xilinx 14.4 ISE

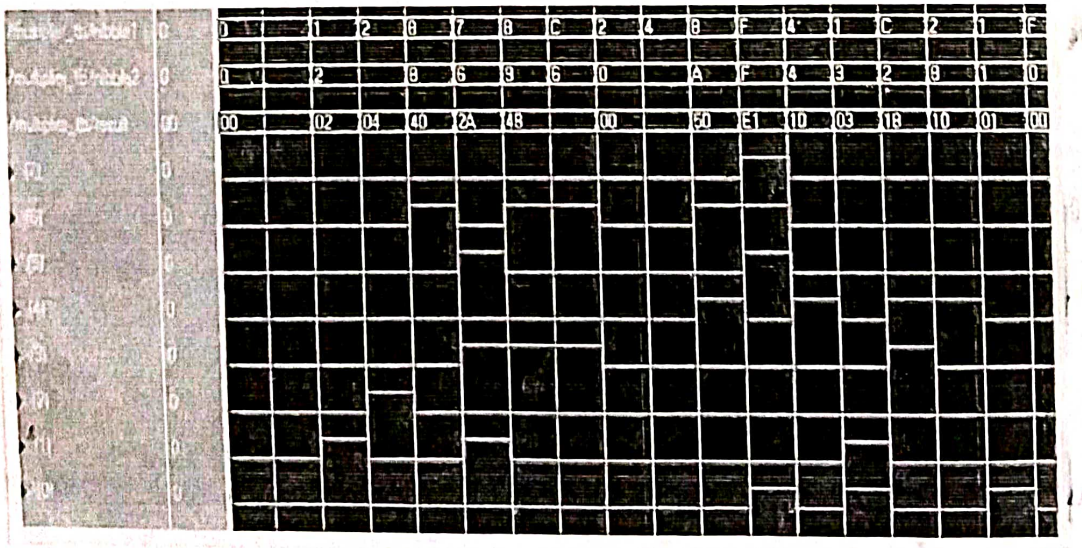
VHDL Code:

```
library IEEE;
use IEEE. STD_LOGIC_1164. ALL;
use IEEE. STD_LOGIC_UNSIGNED. ALL;
entity updown counter is
port ( clk, reset, updown : in STD_LOGIC;
    counter : out STD_LOGIC_VECTOR (3 downto 0));
end updown counter;
architecture behave of updown counter is
signal counter_updown : STD_LOGIC_VECTOR (3 downto 0);
begin
process (clk)
begin
  if (rising_edge (clk)) then
  if ( reset = '1' ) then
  counter_updown <= x "0";
elsif (updown = '1') then
    counter_updown <= counter_updown - x "1";
else
    counter updown <= counter_updown + x "1";
end if;
end process;
counter <= counter_updown;
end behavioral;
```

5/11/23

Teacher's Signature : _____

Time	state	Counter	Product Register	St	M	K	Load	Ad	Sh	Done
A_0	S_0	00	00000000	0	0	0	0	0	0	0
A_1	S_0	00	00000000	1	0	0	1	0	0	0
A_2	S_1	00	00000 1011	0	1	0	0	1	0	0
A_3	S_2	00	01101 1011	0	1	0	0	0	1	0
A_4	S_1	00	00110 1101	0	1	0	0	1	0	0
A_5	S_0	10	10011 1101	0	1	0	0	0	1	0
A_6	S_1	10	10011 1101	0	0	0	0	0	1	0
A_7	S_1	11	01001 1110	0	1	1	0	1	0	0
A_8	S_2	11	00100 1111	0	1	1	0	0	1	0
A_9	S_3	00	01000 1111	0	1	0	0	0	0	1



Date _____

Expt. No. _____ Page No. _____

Aim: To design 4 bit multiplier circuit using add-shift state machine model.

Software Used: Xilinx 14.4 ISE

VHDL Code

```
library IEEE;
use IEEE.NUMERIC_BIT.ALL;
entity MULT is
port( clk, st : in bit;
  mplier, mcand : in unsigned (3 downto 0);
  done : out bit);
end MULT;
architecture behave of MULT is
signal state : integer range 0 to a;
signal acc : unsigned (8 downto 0);
  alias m : bit : 0 acc(0);
begin process (clk) begin
if (clk'event and clk = '1') then
case state is
when 0 => if (st = '1') then acc (8 downto 4)
  <= "00000"; acc(3 downto 0) <= mplier;
  state <= 1; end if;
when 1|3|5|7 =>
if (m = '1') then acc (8 downto 4) <= '0' &
  acc (7 downto 4) + mcand;
    state <= state + 1;
else acc <= '0' & acc (8 downto 1);
```

Teacher's Signature : _____

Date _____

Expt. No. _____ Page No. _____

```
state <= state + 2;
end if;
when 2 | 4 | 6 | 8 =>
acc <= '0' & acc(8 downto 1);
    state <= state + 1;
when a => state <= 0;
end case;
end if;
end process;
done <= '1' when state = 9 else "0";
end behave;
```

5/11/23

Teacher's Signature : _____