EXPERIMENT No-11

Modeling Registers

a)

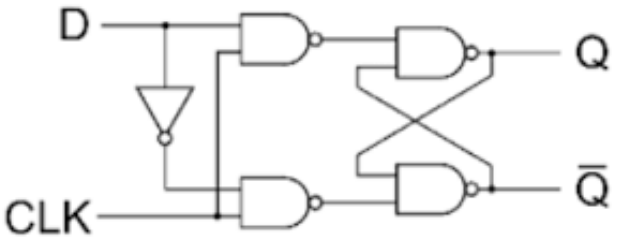
Aim: To write a code in VHDL for implementing the D flip-flop and to verify the functionality.

Description:

Truth table:

Inputs	Outputs	
D	Q	Qb
0	0	1
1	1	0

Circuit Diagram:



RTL Schematic:

VHDL program:

```
library ieee;
use ieee.std_logic_1164.all;
entity d_flipflop is
port(d, clk: in std_logic;
Q: inout std_logic:='0';
Qb: inout std_logic:= '1');
end d_flipflop;
architecture behaviour of d_flipflop is
begin
process(d, clk)
begin
if (clk='0' and clk'event) then
q<=d;
qb<=not(d);
end if;
end process;
end behaviour;
```

Waveforms/Outputs:

b)

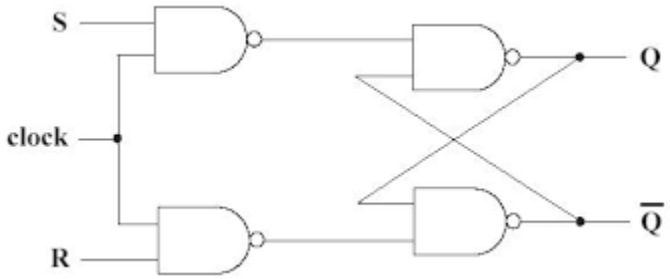
Aim: To write a code in VHDL for implementing the SR flip-flop and to verify the functionality.

Description:

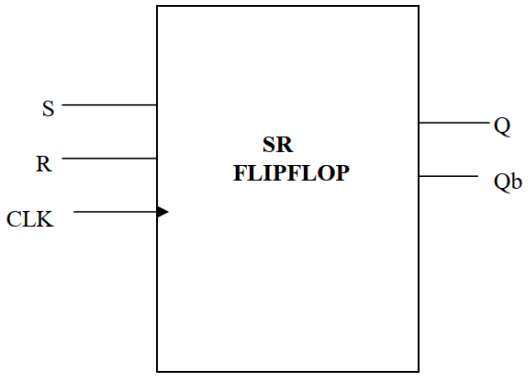
Truth Table:

Inputs		Outputs	
S	R	Q	Qb
0	0	Q	Qb
0	1	0	1
1	0	1	0
1	1	X	X

Circuit diagram:



RTL Schematic:



VHDL Program:

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

```
entity SR is
port(S,R,clk: in std_logic;
        Q:inout std_logic:='0'; Qb:inout std_logic:='1');
end SR;
architecture ff of SR is
begin
process(S,R,clk)
variable t,tb: std_logic;
begin
t:=Q;
tb:=Qb;
if (clk='0'and clk'event) then
if(S='0'and R='0') then t:=t;tb:=tb;
elsif(S='0'and R='1') then t:='0';tb:='1';
elsif(S='1'and R='0') then t:='1';tb:='0';
elsif(S='1'and R='1') then t:='U';tb:='U';
end if;
Q<=t;
Qb<=tb;
end if;

end process;
end ff;
```

Waveforms:

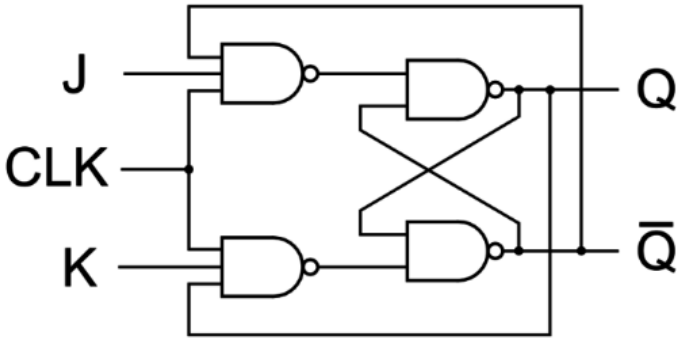
C)

Aim: To write a code in VHDL for implementing the JK flip-flop and to verify the functionality.

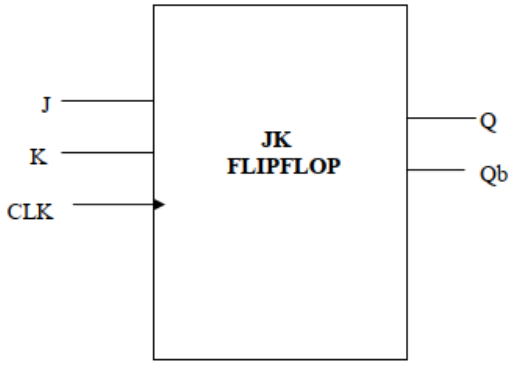
Description:
Truth Table:

Inputs		Outputs	
J	K	Q	Qb
0	0	Q	Qb
0	1	0	1
1	0	1	0
1	1	Q’	Qb’

Circuit diagram:



RTL Schematic:



VHDL Program:

```
library ieee;
use ieee.std_logic_1164.all;
entity JK is
port(J,K,clk: in std_logic;
```

```
Q:inout std_logic:='0';
Qb:inout std_logic:='1');
end JK;
architecture ff of JK is
begin
process(J,K,clk)
variable t,tb: std_logic;
begin
t:=Q;
tb:=Qb;
if (clk='0'and clk'event) then
if(J='0'and K='0') then t:=t;tb:=tb;
elsif(J='0'and K='1') then t:='0';tb:='1';
elsif(J='1'and K='0') then t:='1';tb:='0';
elsif(J='1'and K='1') then t:=not t; tb:=not tb;
end if;
end if;
Q<=t;
Qb<=tb;
end process;
end ff;
```

Waveforms:

d)

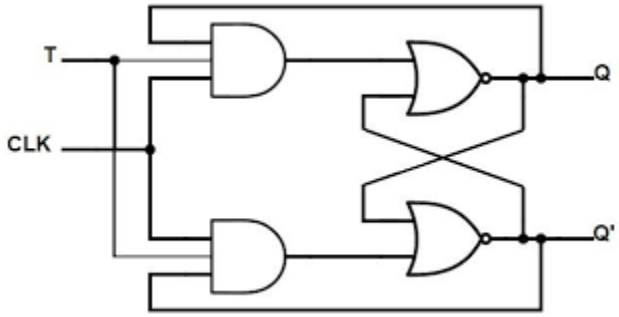
Aim: To write a code in VHDL for implementing the T flip-flop and to verify the functionality.

Description:

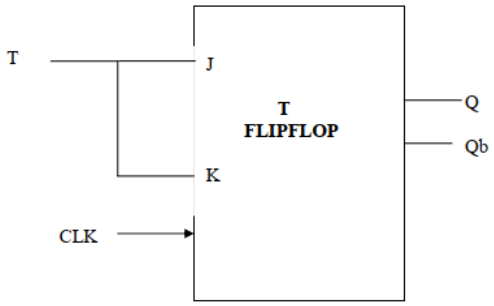
Truth Table:

Inputs	Outputs	
T	Q	Qb
0	Q	Qb
1	Qb	Q

Circuit diagram:



RTL Schematic:



VHDL Program:

```
library ieee;
use ieee.std_logic_1164.all;
entity T is
port(T,clk: in std_logic;
Q:inout std_logic:='0';
Qb:inout std_logic:='1');
end T;
architecture ff of T is
```

```
component JK
port(J,K,clk: in std_logic;
Q:inout std_logic:='0';
Qb:inout std_logic:='1');
end component;
begin
X1: JK port map(T,T,clk,Q,Qb);
end ff;
```

Waveforms: