# EXP-05 Implementation-of-flipflops-using-verilog

## Aim:

To implement all the flipflops using verilog and validating their functionality using their functional tables

## Equipments Required:

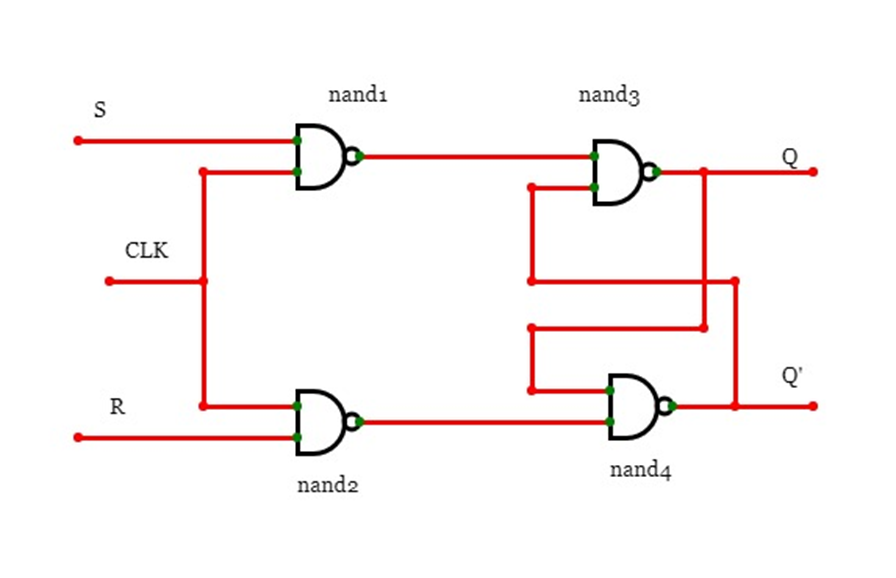
Hardware – PCs, Cyclone II , USB flasher

Software – Quartus prime

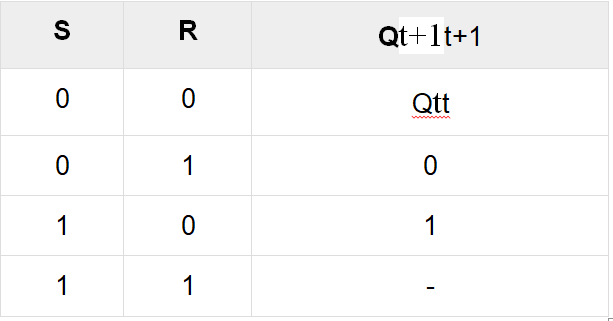
## Theory:

### SR Flip-Flop:

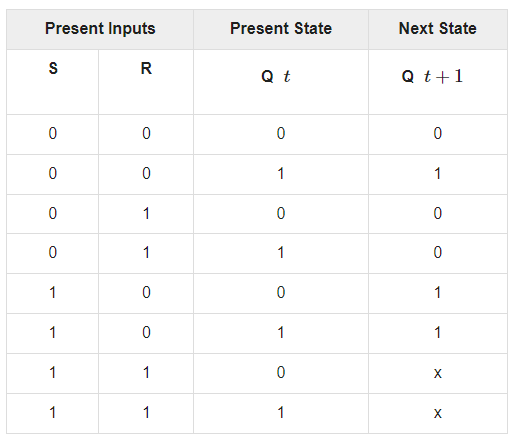
SR flip-flop operates with only positive clock transitions or negative clock transitions. Whereas, SR latch operates with enable signal. The circuit diagram of SR flip-flop is shown in the following figure.

[](https://user-images.githubusercontent.com/36288975/167910294-bb550548-b1dc-4cba-9044-31d9037d476b.png)

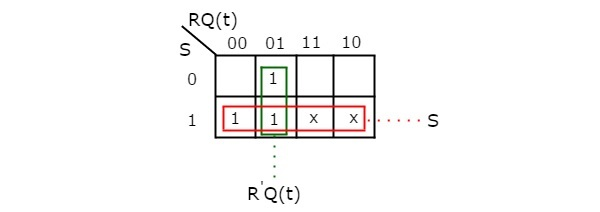
This circuit has two inputs S & R and two outputs Qtt & Qtt’. The operation of SR flipflop is similar to SR Latch. But, this flip-flop affects the outputs only when positive transition of the clock signal is applied instead of active enable. The following table shows the state table of SR flip-flop.

[](https://user-images.githubusercontent.com/36288975/167910648-ced88e69-869c-42e2-9718-a285a3902446.png)

Here, Qtt & Qt+1t+1 are present state & next state respectively. So, SR flip-flop can be used for one of these three functions such as Hold, Reset & Set based on the input conditions, when positive transition of clock signal is applied. The following table shows the characteristic table of SR flip-flop. Present Inputs Present State Next State

[](https://user-images.githubusercontent.com/36288975/167908180-5fc9d589-1cb5-41f5-b2c8-927e04f5f387.png)

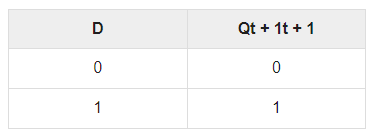
By using three variable K-Map, we can get the simplified expression for next state, Qt+1t+1. The three variable K-Map for next state, Qt+1t+1 is shown in the following figure.

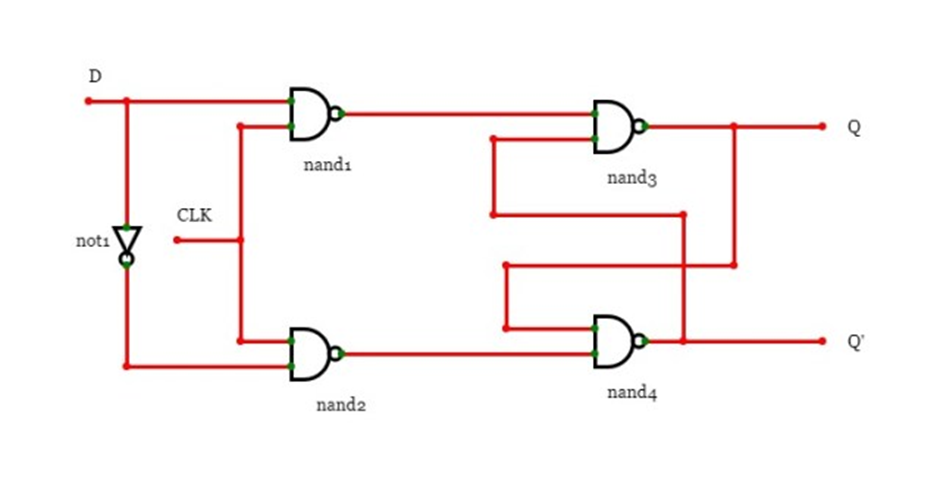
[](https://user-images.githubusercontent.com/36288975/167908214-25b30a54-db20-4bcb-9385-5f93a1982a09.png)

The maximum possible groupings of adjacent ones are already shown in the figure. Therefore, the simplified expression for next state Qt+1t+1 is Q(t+1)=S+R′Q(t)Q(t+1)=S+R′Q(t)

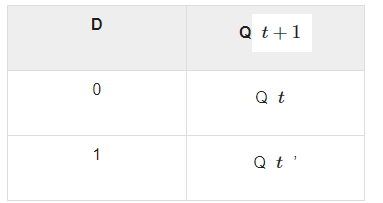
### D Flip-Flop:

D flip-flop operates with only positive clock transitions or negative clock transitions. Whereas, D latch operates with enable signal. That means, the output of D flip-flop is insensitive to the changes in the input, D except for active transition of the clock signal. The circuit diagram of D flip-flop is shown in the following figure.

This circuit has single input D and two outputs Qtt & Qtt’. The operation of D flip-flop is similar to D Latch. But, this flip-flop affects the outputs only when positive transition of the clock signal is applied instead of active enable. The following table shows the state table of D flip-flop. [](https://user-images.githubusercontent.com/36288975/167908342-e03f0cbb-5958-43bb-b74a-5e3ec2341675.png)

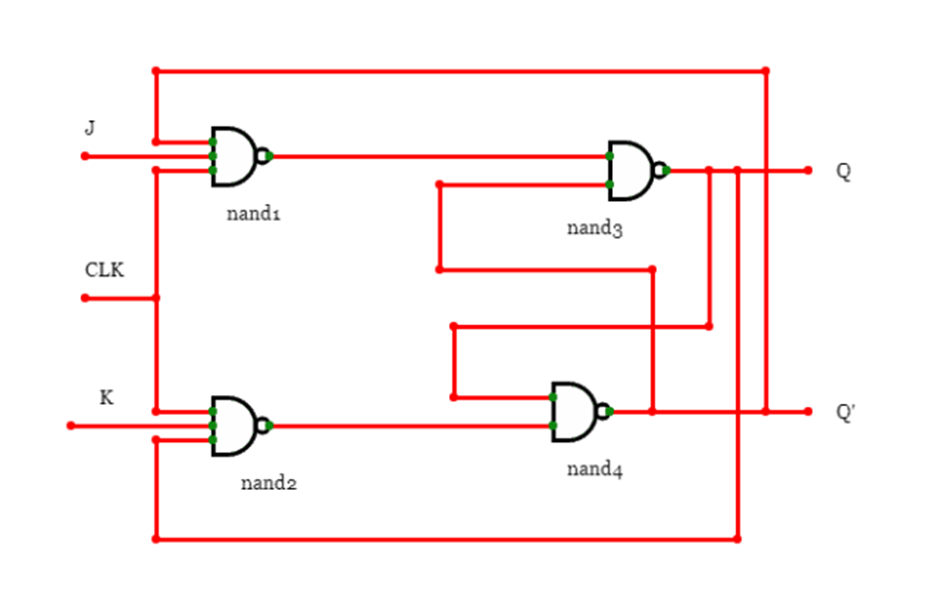
[](https://user-images.githubusercontent.com/36288975/167910325-aeef0739-0a54-40e2-bebd-6f5fa0cad10e.png)

Therefore, D flip-flop always Hold the information, which is available on data input, D of earlier positive transition of clock signal. From the above state table, we can directly write the next state equation as Qt+1t+1 = D

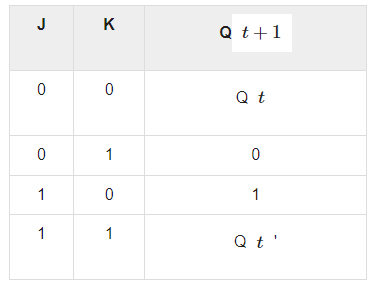
[](https://user-images.githubusercontent.com/36288975/167908850-d39d07ba-7f9d-490a-b9f2-274e189fd047.png)

Next state of D flip-flop is always equal to data input, D for every positive transition of the clock signal. Hence, D flip-flops can be used in registers, shift registers and some of the counters.

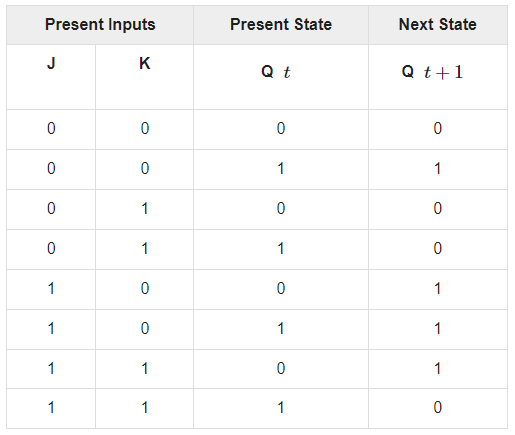
### JK Flip-Flop:

JK flip-flop is the modified version of SR flip-flop. It operates with only positive clock transitions or negative clock transitions. The circuit diagram of JK flip-flop is shown in the following figure. [](https://user-images.githubusercontent.com/36288975/167910378-d2d984a7-2815-4d17-8c41-ee4bdf59ec24.png)

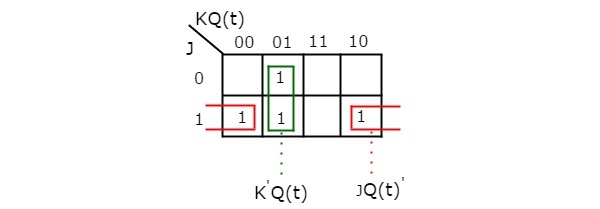
This circuit has two inputs J & K and two outputs Qtt & Qtt’. The operation of JK flip-flop is similar to SR flip-flop. Here, we considered the inputs of SR flip-flop as S = J Qtt’ and R = KQtt in order to utilize the modified SR flip-flop for 4 combinations of inputs. The following table shows the state table of JK flip-flop.

[](https://user-images.githubusercontent.com/36288975/167908575-59c35afb-50d3-46a2-888c-47478a3179d5.png)

Here, Qtt & Qt+1t+1 are present state & next state respectively. So, JK flip-flop can be used for one of these four functions such as Hold, Reset, Set & Complement of present state based on the input conditions, when positive transition of clock signal is applied. The following table shows the characteristic table of JK flip-flop. Present Inputs Present State Next State

[](https://user-images.githubusercontent.com/36288975/167908664-c854ffe9-0bd3-44c2-bfa6-e53928181c69.png)

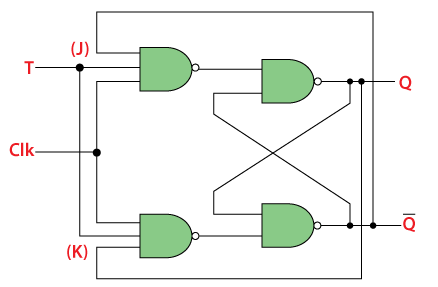
By using three variable K-Map, we can get the simplified expression for next state, Qt+1t+1. Three variable K-Map for next state, Qt+1t+1 is shown in the following figure.

[](https://user-images.githubusercontent.com/36288975/167908688-fa93c3e9-8323-4864-947d-c11d163d5a90.png)

The maximum possible groupings of adjacent ones are already shown in the figure. Therefore, the simplified expression for next state Qt+1t+1 is Q(t+1)=JQ(t)′+K′Q(t)Q(t+1)=JQ(t)′+K′Q(t)

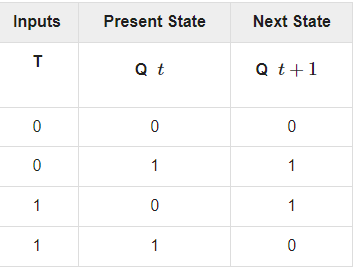
### T Flip-Flop:

T flip-flop is the simplified version of JK flip-flop. It is obtained by connecting the same input ‘T’ to both inputs of JK flip-flop. It operates with only positive clock transitions or negative clock transitions. The circuit diagram of T flip-flop is shown in the following figure.

[](https://user-images.githubusercontent.com/36288975/167911534-5f3c445d-bc68-46e2-9a9c-7efce5febc60.png)

This circuit has single input T and two outputs Qtt & Qtt’. The operation of T flip-flop is same as that of JK flip-flop. Here, we considered the inputs of JK flip-flop as J = T and K = T in order to utilize the modified JK flip-flop for 2 combinations of inputs. So, we eliminated the other two combinations of J & K, for which those two values are complement to each other in T flip-flop. The following table shows the state table of T flip-flop.

Here, Qtt & Qt+1t+1 are present state & next state respectively. So, T flip-flop can be used for one of these two functions such as Hold, & Complement of present state based on the input conditions, when positive transition of clock signal is applied. The following table shows the characteristic table of T flip-flop. Inputs Present State Next State

[](https://user-images.githubusercontent.com/36288975/167909015-53aa9450-3f28-4202-887a-79d88228f8a0.png)

From the above characteristic table, we can directly write the next state equation as Q(t+1)=T′Q(t)+TQ(t)′ ⇒Q(t+1)=T⊕Q(t)

## Procedure:

1.Using nand gates and wires construct sr flip flop.

2.Repeat same steps to construct JK,D,T flipflops.

3.Find Rtl logic and timing diagram for all flipflops.

4.end the program

## Program:

### SR Flip-Flop:

module SR\_flipflop(S,R,clk,Q,Qbar);

input S,R,clk;

output Q,Qbar;

wire X,Y;

nand (X,S,clk);

nand (Y,R,clk);

nand (Q,X,Qbar);

nand (Qbar,Y,Q);

endmodule

### D Flip-Flop:

module D\_flipflop(D,clk,Q,Qbar);

input D,clk;

output Q,Qbar;

assign Dbar=~D;

wire X,Y;

nand (X,D,clk);

nand (Y,Dbar,clk);

nand (Q,X,Qbar);

nand (Qbar,Y,Q);

endmodule

### JK Flip-Flop:

module JK\_flipflop(J,K,clk,Q,Qbar);

input J,K,clk;

output Q,Qbar;

wire X,Y;

nand (X,J,clk,Qbar);

nand (Y,K,clk,Q);

nand (Q,X,Qbar);

nand (Qbar,Y,Q);

endmodule

### T Flip-Flop:

module T\_flipflopo(T,clk,Q,Qbar);

input T,clk;

output Q,Qbar;

wire S,R;

nand (S,T,clk,Qbar);

nand (R,T,clk,Q);

nand (Q,S,Qbar);

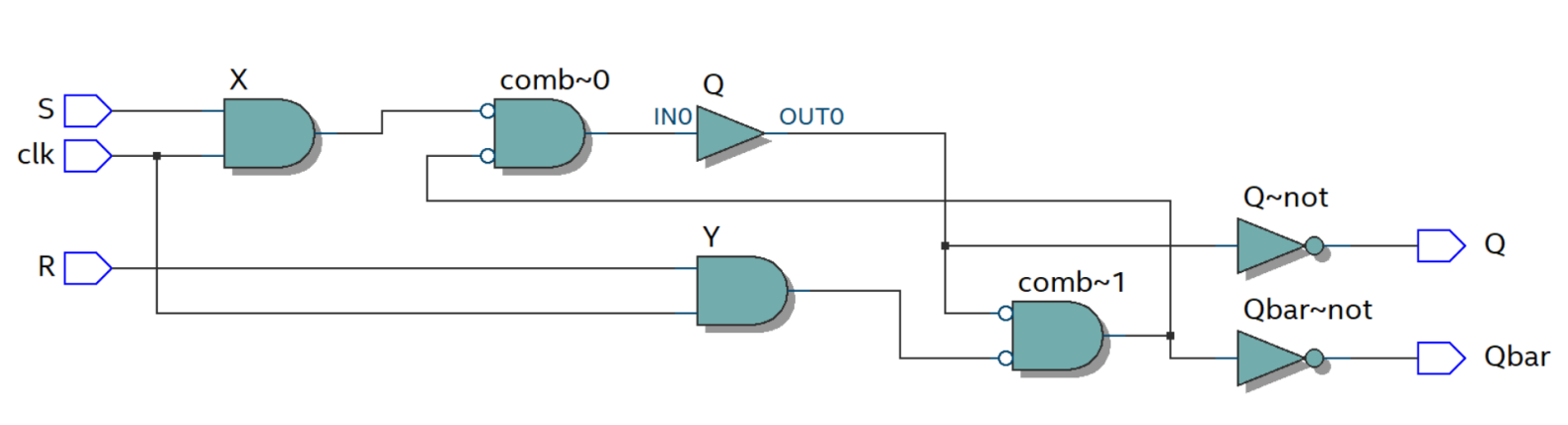
nand (Qbar,R,Q);

endmodule

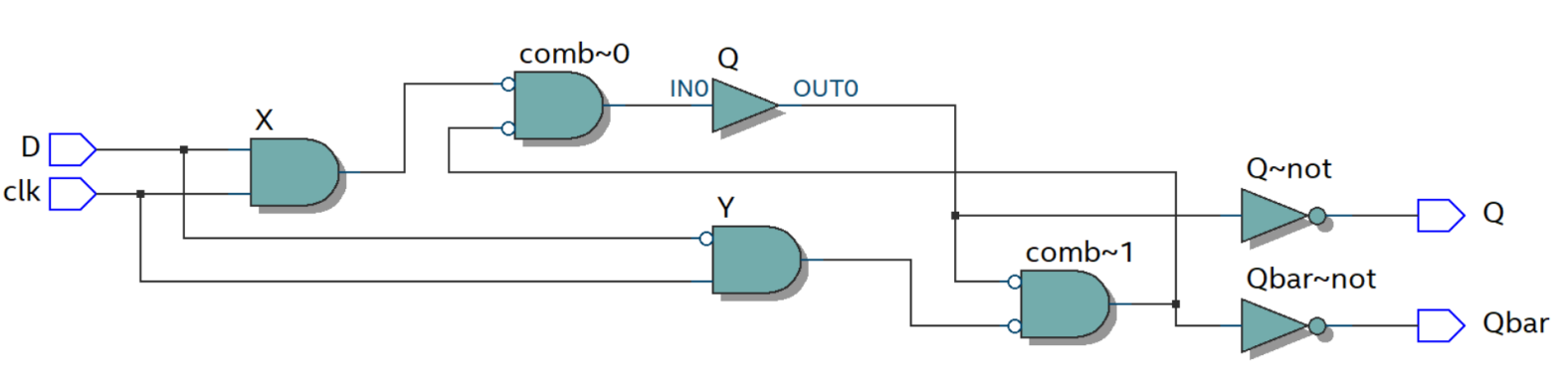
## Output:

### RTL logic for Flip-Flops :

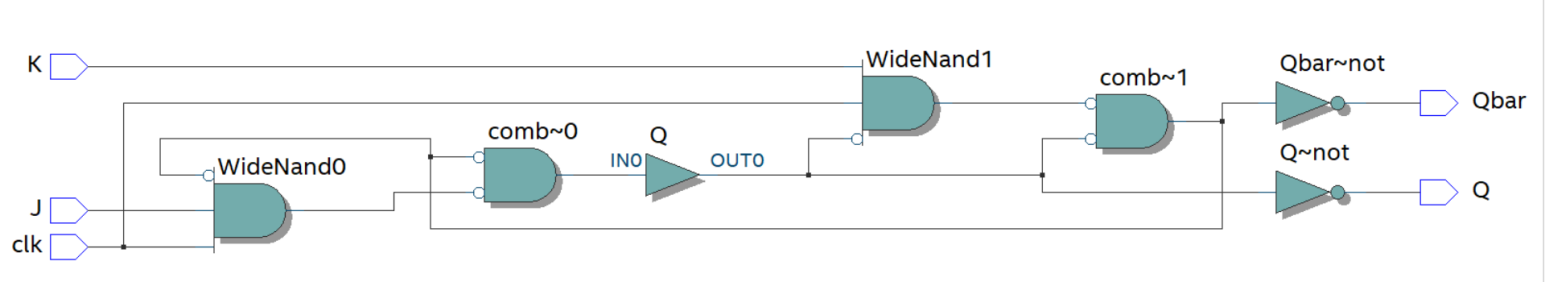
#### SR Flip-Flop:

[](https://private-user-images.githubusercontent.com/148048602/291846617-1860e222-2bf6-4c58-a28f-efcde7adcc06.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ2NjE3LTE4NjBlMjIyLTJiZjYtNGM1OC1hMjhmLWVmY2RlN2FkY2MwNi5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT05MDEzNzYyZDlhZTNhM2JmZjJmMGRhY2I1MTVjNDQ2ODIzMTc4ZDFlMTJiNjBkYjNmY2E3MGVmNTliZjZkMDA2JlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9.NAvrrAnQ1UZCYTB3IjxUEKTytlUbk5lQnoEXfC1AJVk)

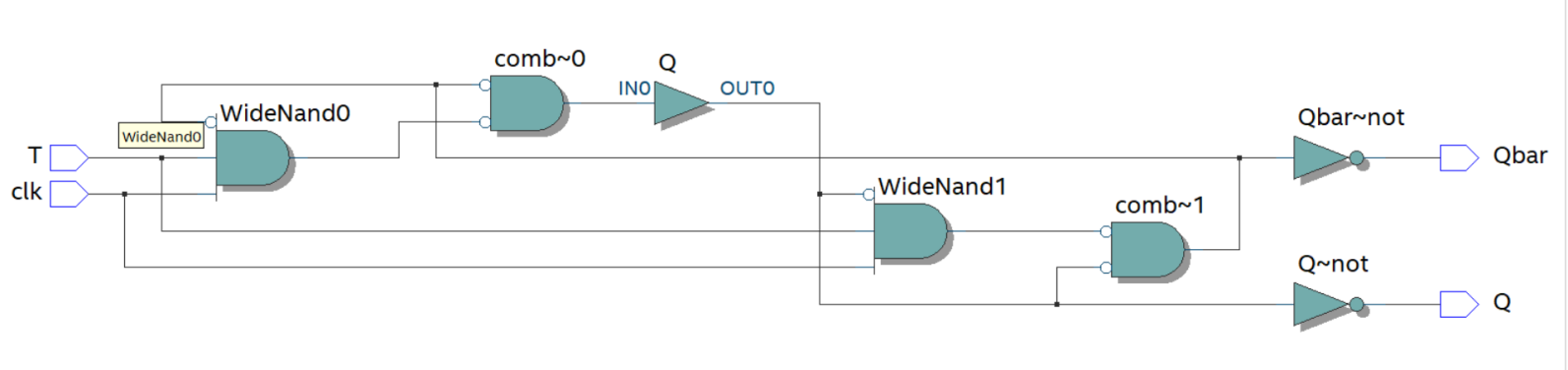
#### D Flip-Flop:

[](https://private-user-images.githubusercontent.com/148048602/291847044-6463cd59-e74b-4f4d-a5dd-7d21f601e3ae.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ3MDQ0LTY0NjNjZDU5LWU3NGItNGY0ZC1hNWRkLTdkMjFmNjAxZTNhZS5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT0yMzhkOTI0NDNlMjJlNzlkYjYwMzM3NTdkOGE5MDUzY2QzZWM4OTQyM2E5Y2RlNGUwMTRlZTBmOTRjMTFlMmI1JlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9.2C85D5pLNc5WBZkXGbTTtE4jQ8Hhaox36-zgHDYnbm4)

#### JK Flip-Flop:

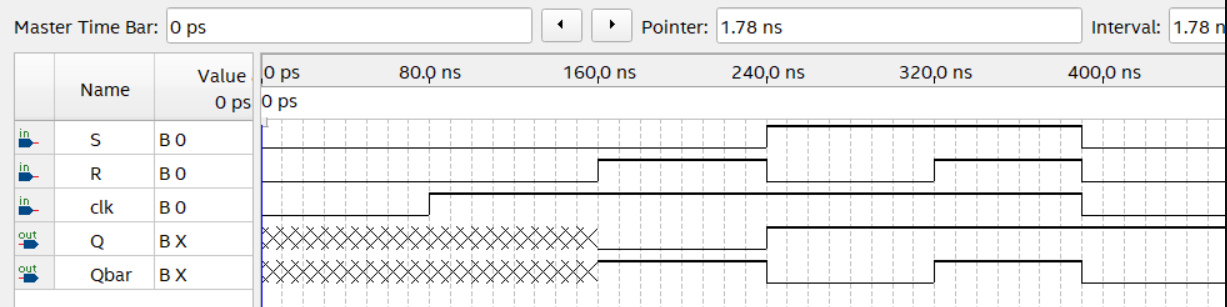
[](https://private-user-images.githubusercontent.com/148048602/291847120-65bd2f58-c125-45a5-a85d-ed82fe790869.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ3MTIwLTY1YmQyZjU4LWMxMjUtNDVhNS1hODVkLWVkODJmZTc5MDg2OS5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT01NzdiZmQ1YWU3ODBhZDhmMTMyYTc4YmI2NDE3ZmFjMDc4NDIwYzcwZDZlZTUxODY0ODZhYTdlOGZhZjg0MzYyJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9.SPi7iUg2h1RTOejN8OPPQo0vgonQ9JRNAaB6YVyVZ4s)

#### T Flip-Flop:

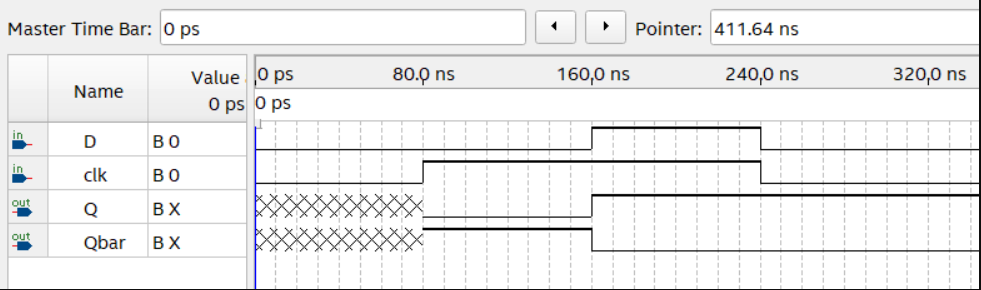
[](https://private-user-images.githubusercontent.com/148048602/291846984-30a0ef04-ebb4-4bb9-b19b-ad31bcfed389.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ2OTg0LTMwYTBlZjA0LWViYjQtNGJiOS1iMTliLWFkMzFiY2ZlZDM4OS5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT02YTA2NTBkNDU2NDhmMjk2MDFmOTUwZWQ0NTYzMGY0YWE2ODg5MGMxNjFkOGYxZTBkYTY3YjlmNzc2YjcwNWRjJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9.VS-vu2JEAoCrCndyhLIs3YG3S1WWsa5azkQn1WIBxa8)

### Timing diagrams for Flip-Flops:

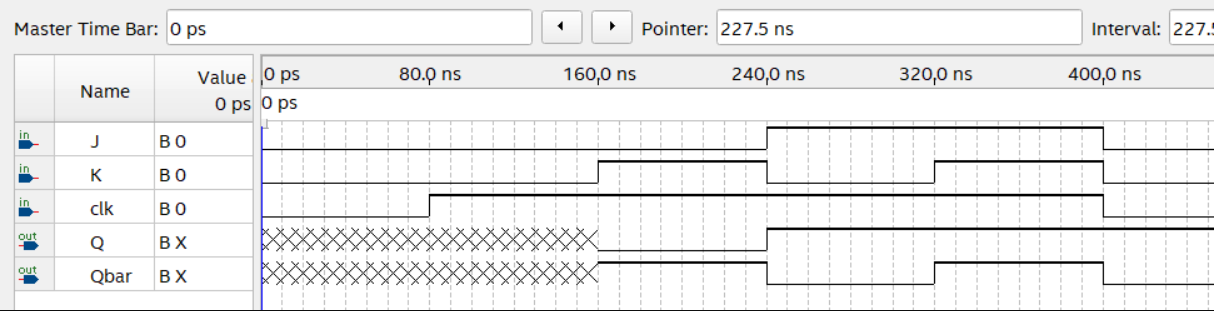
#### SR Flip-Flop:

[](https://private-user-images.githubusercontent.com/148048602/291847892-95beb77f-4e53-42a9-a953-60d4cc3ee2f0.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ3ODkyLTk1YmViNzdmLTRlNTMtNDJhOS1hOTUzLTYwZDRjYzNlZTJmMC5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT00ZThhZmE2NGRlOGM2MGJjYmE5NjE4ZjI0NGJjOWQ1MGZmMjQxNTgxY2RkMTQ5MmE0OGNhNDI4ZThhZDYxZGJkJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9.NFmYHNAhKBn1itDyQmZj7NU38pol5uoN7mfjyU4kUo4)

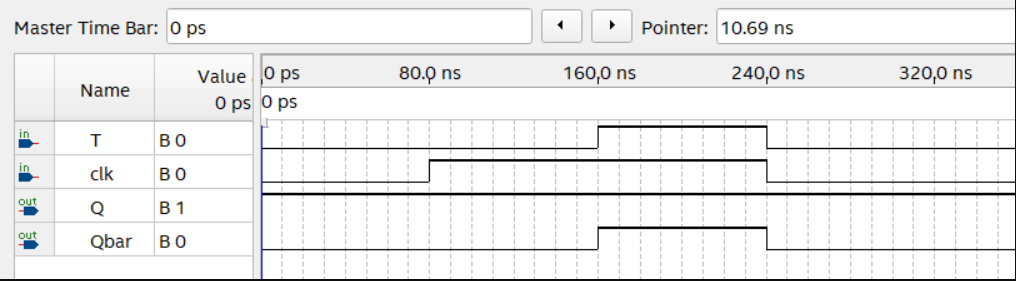
#### D Flip-Flop:

[](https://private-user-images.githubusercontent.com/148048602/291847799-28ec687f-81a2-4fdf-a02c-f8c72336f43b.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ3Nzk5LTI4ZWM2ODdmLTgxYTItNGZkZi1hMDJjLWY4YzcyMzM2ZjQzYi5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT0wMzg4MWIxNzQ3NDM0NjI2MDMyNjlhNmZhOTcyZmRkYWVmZDdjZjQ0YzQ3Zjc5NWMxODg2M2QwNTM4ZmQ1YTJjJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9.O7FQK3pxmt7YwpkUZLEFVRvjFXXIoJ-SFwe0zc9pYrc)

#### JK Flip-Flop:

[](https://private-user-images.githubusercontent.com/148048602/291847857-8a26a505-1b36-463c-b502-2aec73c996a3.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ3ODU3LThhMjZhNTA1LTFiMzYtNDYzYy1iNTAyLTJhZWM3M2M5OTZhMy5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT0yZGM1MTdlYjkxMWE5ODZhMmI4NjZhNTU0OTRmNWVhNWNmZDA1OGE2ODU4OGEwYTE0NzRhNjI3ZDQxMjA4NjMyJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9._JpF5p95-Q9ATWY5QXzbG2P_gZUKuTuNDJC9mJWWCIA)

#### T Flip-Flop:

[](https://private-user-images.githubusercontent.com/148048602/291847683-de2a903d-6b2e-493a-a8cc-488f7d3293bc.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTEiLCJleHAiOjE3MDMzMzI5MDgsIm5iZiI6MTcwMzMzMjYwOCwicGF0aCI6Ii8xNDgwNDg2MDIvMjkxODQ3NjgzLWRlMmE5MDNkLTZiMmUtNDkzYS1hOGNjLTQ4OGY3ZDMyOTNiYy5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBSVdOSllBWDRDU1ZFSDUzQSUyRjIwMjMxMjIzJTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDIzMTIyM1QxMTU2NDhaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT0xZmNlYTcxMTVjYzU1MDY0NjA1YjE2MmQ3ZjFkMjQ0ZDE2NWIxNzdmYjlkNjc0NTRjZDdkNTY1N2YxYTllYmE3JlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCZhY3Rvcl9pZD0wJmtleV9pZD0wJnJlcG9faWQ9MCJ9.FfFZULXxF9yksiZGnh3t0fdg-hpufVhn2Sn6tOOaapE)

## Result:

Thus implementation of SR, JK, D and T flipflops using nand gates are done sucessfully.