EXP 5 :JK FLIPFLOP USING IF-ELSE

AIM:

To implement JK flipflop using verilog and validating their functionality using their functional tables

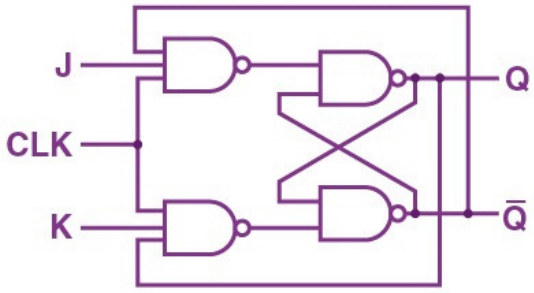
SOFTWARE REQUIRED:

Quartus prime

THEORY;

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.



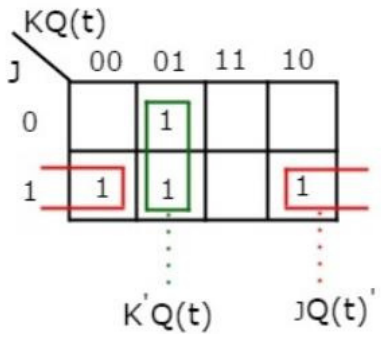
This circuit has two inputs J & K and two outputs Qtt & Qtt'. The operation of JK flip-flop is similar to SR flip-flop. Here, we considered the inputs of SR flip-flop as S = J Qtt' and R = KQtt in order to utilize the modified SR flip-flop for 4 combinations of inputs. The following table shows the state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

J	K	Q $t+1$
0	0	Q t
0	1	0
1	0	1
1	1	Q t '

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

Present Inputs		Present State	Next State
J	K	Q t	Q $t+1$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

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.



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)

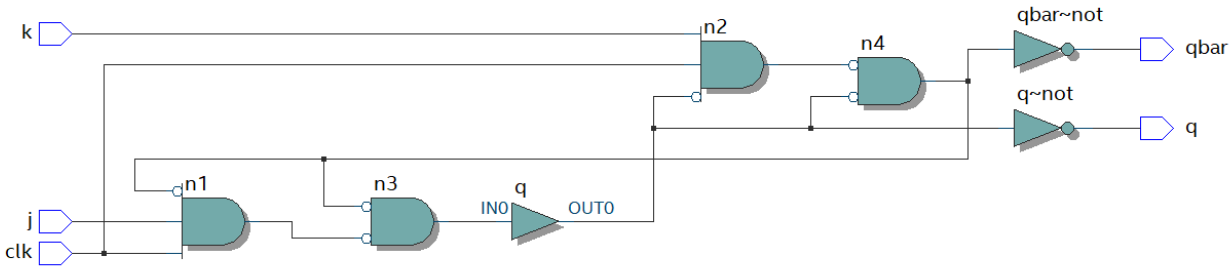
Procedure

1. Type the program in Quartus software.
2. Compile and run the program.
3. Generate the RTL schematic and save the logic diagram.
4. Create nodes for inputs and outputs to generate the timing diagram.
5. For different input combinations generate the timing diagram.

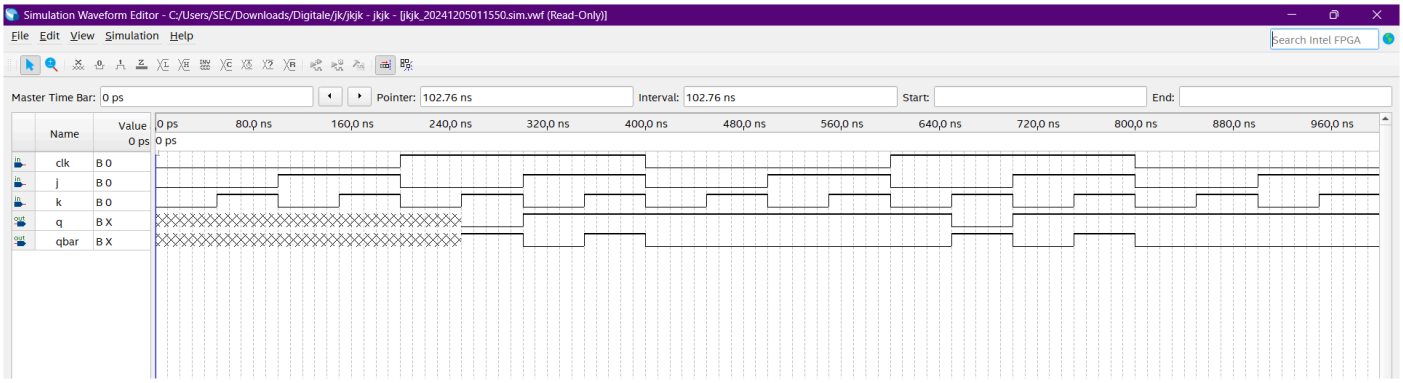
PROGRAM:

```
module jkjk (
    input j,        // J input
    input k,        // K input
    input clk,      // Clock input
    output q,       // Q output
    output qbar     // Complement of Q
);

    wire x, y, w, z;

    // Internal feedback connections
    assign w = q;
    assign z = qbar;

    // NAND gate logic for JK flip-flop
    nand n1(x, z, j, clk);  // x = ~(z & j & clk)
    nand n2(y, k, w, clk);  // y = ~(k & w & clk)
    nand n3(q, x, z);       // q = ~(x & z)
    nand n4(qbar, y, w);    // qbar = ~(y & w)

endmodule
```

RTL LOGIC FOR FLIPFLOPS:



TIMING DIGRAMS FOR FLIP FLOPS:



RESULTS:

The JK flip-flop implemented in Verilog has been successfully completed. Its functionality has been validated by comparing its output behavior with the expected values from the functional table. Simulation results confirm that the JK flip-flop operates correctly, exhibiting the characteristic behavior of a flip-flop where the output state depends on both the present state and the input signals, as specified by the JK flip-flop's functional table.

Developed by: Rhudhresh

Register Number: 212223050039