

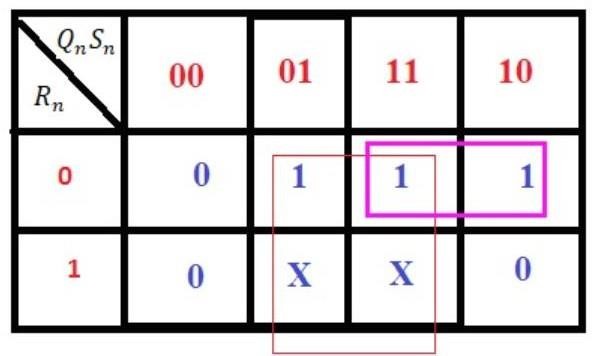
Lab 8: Sequential Circuits | SR Latches and Flipflops

Q 1. An SR latch (Set/Reset) is an asynchronous device: it works independently of control signals and relies only on the state of the S and R inputs.

1. **Generate the Boolean expression for the S-R latch from the truth table given in Table-1.**

Truth table:-

*Figure 1:1.1\_TruthTable of SR latch*



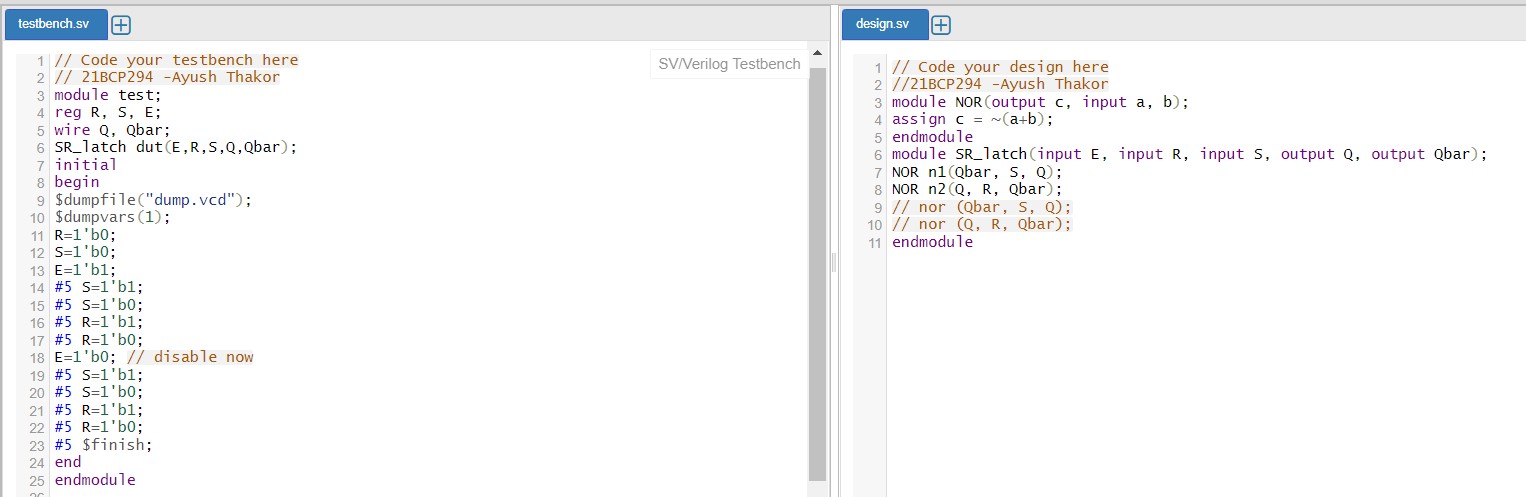
Boolean expression:-

Q(n+1)=Sn+Qn (Rn)’

1. **Write a module for NOR gate and develop a structural verilog code for the S-R latch using the NOR gate module.**

Design code:- Structural code:-

1. **Validate the code via a suitable Testbench code.**



*Figure 2:1.2\_SRlatch\_structural\_code*

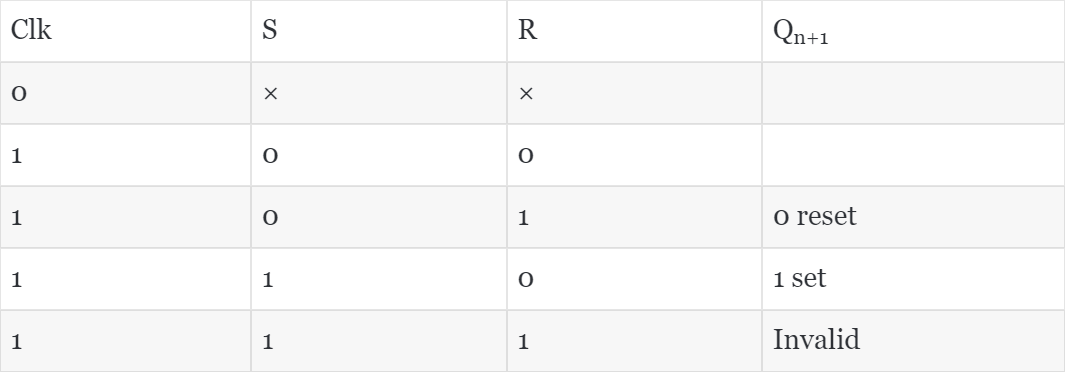
Wave:



*Figure 3:1.2\_SRlatch\_waveform*

Q 2. In S-R latch we do not use a clock. Now if we add an additional clock at input so that the S and R inputs are active only when the clock is high. When the clock goes low, the state of flip-flop is latched and cannot change until the clock goes high again. This clocked addition in S-R latch is also called the S-R flip flop.

1. **Generate the Boolean expression for the S-R flip flop using K-map and truth table given in Table 2**

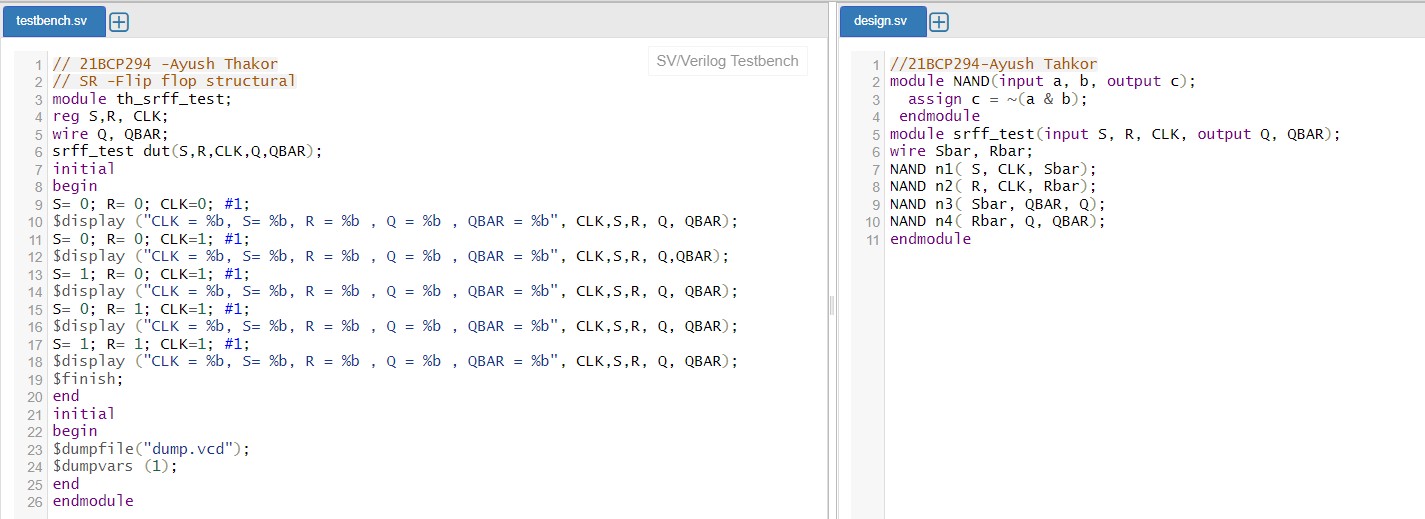
Truth table:-

Boolean expression:-

Q(n+1)=Sn+Qn (Rn)’

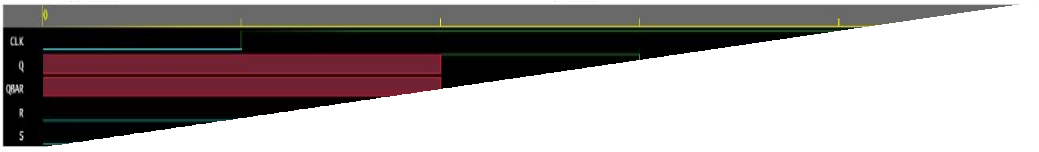
1. **Write a verilog module for NAND gate and utilize it to develop a structural verilog module for S-R flip flop as per figure 3.**

Structural Code:



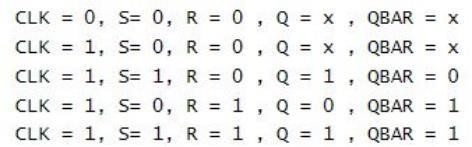
*Figure 4:2.1\_SRFLIPFLOP\_STRUCTURAL*

Wave:-



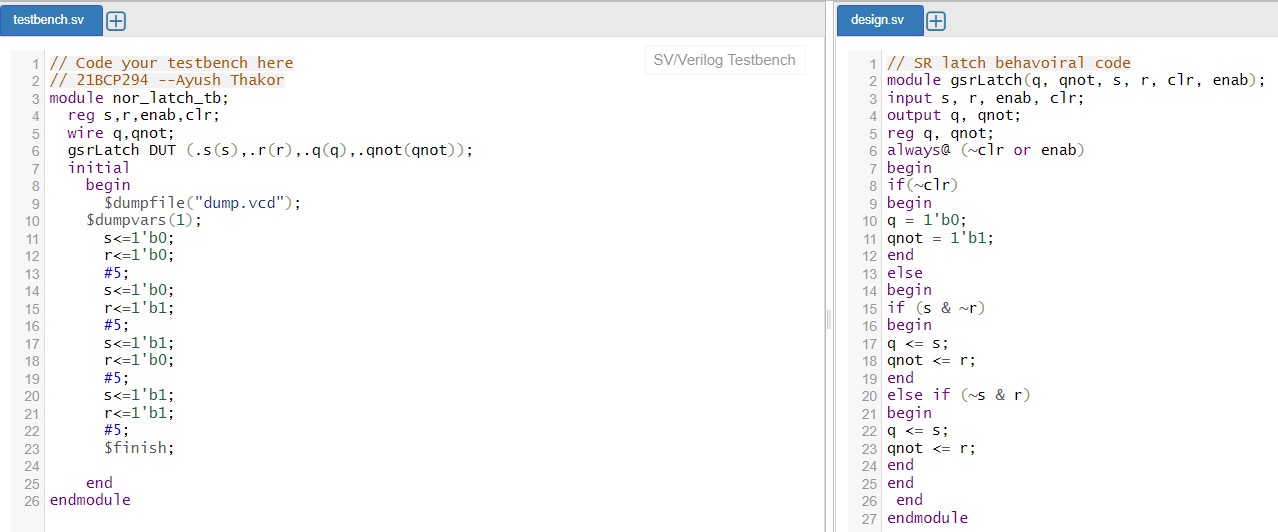
*Figure 5:2.2\_SRflipflop\_waveform*

Output:-



Q 3. We can also develop a behavioral modeling based verilog code for the S-R latch. Here we can use the if-else logic to assign values for output based on the input conditions.

1. **Develop a behavioral verilog code using if-else statements for S-R latch.**
2. **Varify it with a suitable testbench code.**

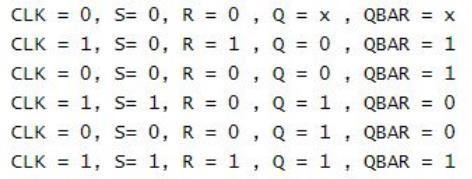


*Figure 6:3.1\_SRlatch\_behavioral*

Wave:-

*Figure 7:3.2\_waveform*

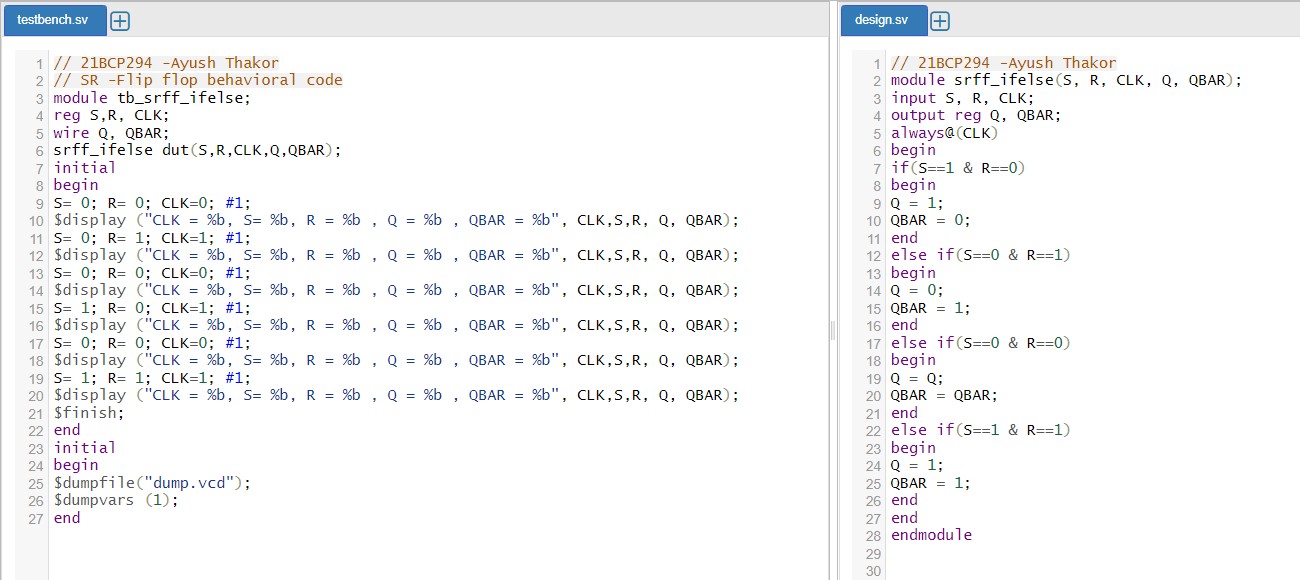
Output:-



*Figure 8:3.3\_output*

Q 4. Develop a similar behavioral code and test bench for S-R flip flop using if-else condition as per question 3.

Behavioral code:-



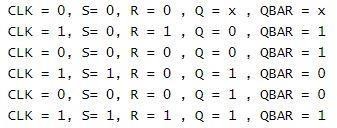
*Figure 9:4.1\_SRflipflop\_behavioral*

Wave:-

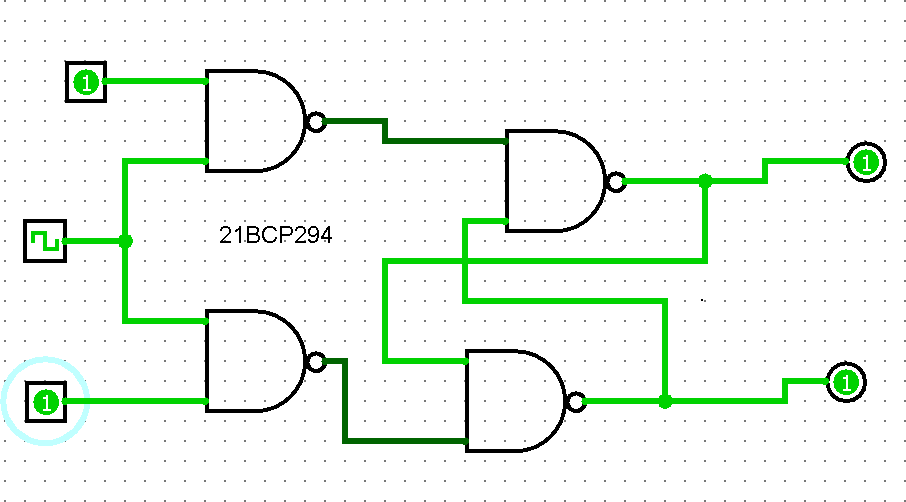


*Figure 10:4.2\_waveform*

Output:-



*Figure 11:4.3\_output*

Q 5.Develop S-R flip flop using logisim Circuit diagram:-

*Figure 12:5.1\_SRflipflop\_logisim*

**Truth Table:**



**Table 8.2 SR Flip-Flop Truth Table**