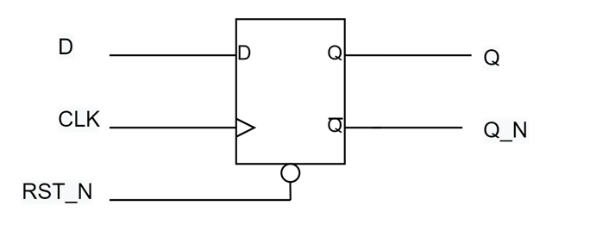
**Design of shift registers**

Flip-flops are primitive components of a shift register or any sequential logic. Figure 1 shows the symbol of a D-type flip,-flop and how it works. Flip-flops are sensitive to the edge of a clock. The reset signal is usually active low, means that when it is in Low (‘0’) logic the flip-flip is in reset state.

At reset state, the output Q is in Low (‘0’) position and Q\_N, which is NOT of the Q is in High (‘1’) position.

The reset is asynchronous to the clock. Means that when active (‘0’), the outputs will go the reset state regardless of clock position.

When RESET\_N pin is not active, i.e. in High (‘1’) position, the output Q gets the value of the input D at the rising edge of the clock and holds this value until updated at the rising edge of the next clock.



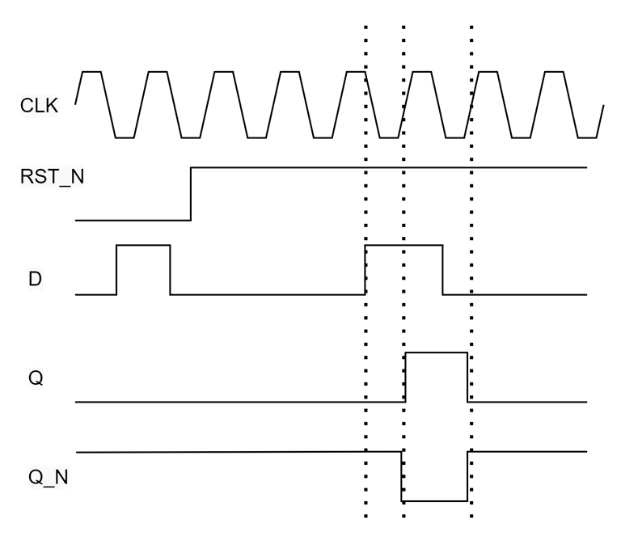


Figure 1: D-type flip-flop

The Verilog description of the above flip-flop is shown in Block 1.

Block 1: D flip-flop Verilog code

**module** dff

(

**input** i\_d,

**input** i\_clk,

**input** i\_rst\_n,

**output** o\_q

);

**reg** r\_q;

**assign** o\_q = r\_q;

**always**@(**posedge** i\_clk **or** **negedge** i\_rst\_n)

**begin**

**if**(!i\_rst\_n)

r\_q<= 0;

**else**

r\_q<=i\_d;

**end**

**end**

**endmodule**

Block 2, suggests a test-bench for the flip-flop

`timescale 1ns/1ns

**module** dff\_tb();

**reg** r\_d, r\_clk, r\_rst\_n;

**wire** w\_q;

**integer** i;

dff uut(

**.**i\_d(r\_d),

**.**i\_clk(r\_clk),

**.**i\_rst\_n(r\_rst\_n),

**.**o\_q(w\_q)

);

**initial**

**begin**

r\_clk = 0;

r\_rst\_n = 0;

r\_d = 0;

#10 r\_rst\_n = 1;

**end**

**always** #10 r\_clk <= ~r\_clk;

**initial**

**begin**

**for**( i= 0; i< 10; i=i+1)

#24r\_d <= ~r\_d;

**end**

**endmodule**

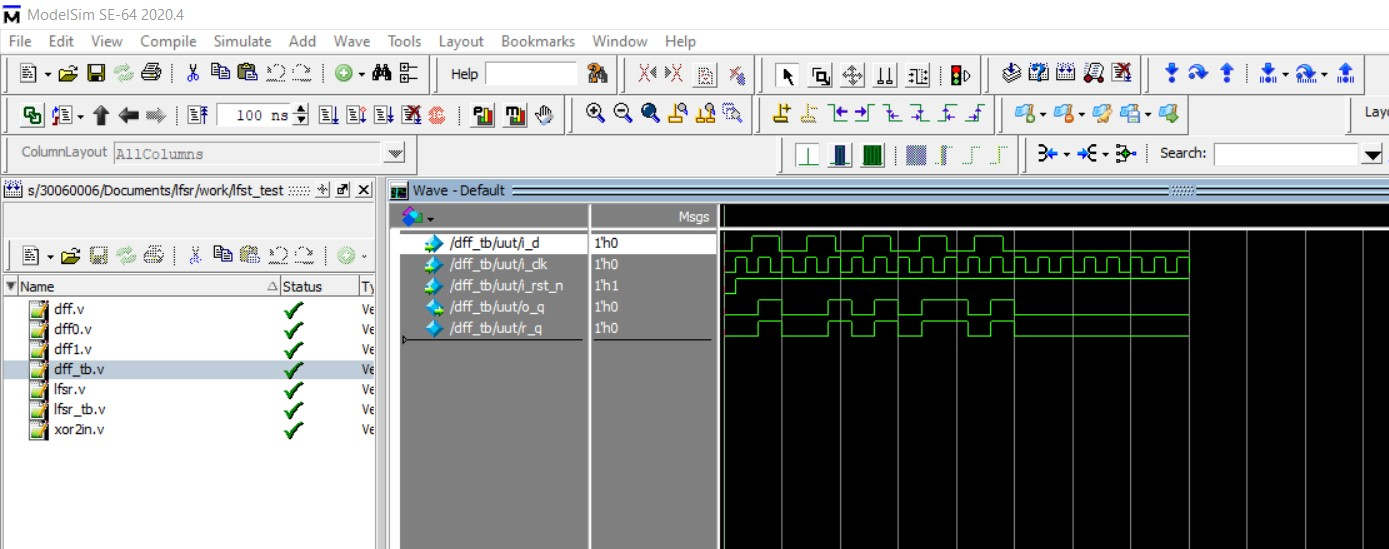


Figure 3: ModelSim simulation of the D flip-flop

Now, if we connect the clock and reset of two flip-flops together and the output of the first one to the input of the next, we will have a two bits shift register (Figure 2). The input D, will appear at the output Q2 of the second flip-flop after two clock pulses (two rising edge of the clock). Why?

Draw the waveform for the shift register in Figure 2.

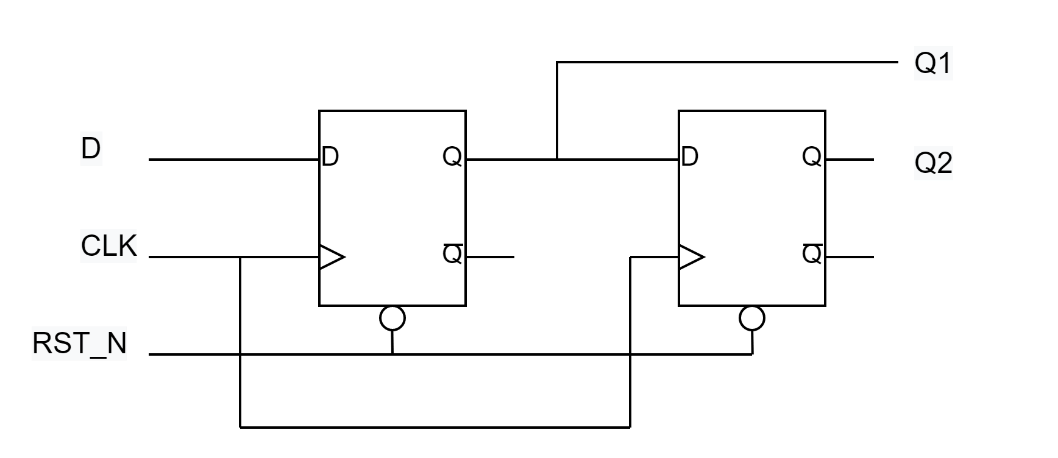


Figure 2: A 2-bit shift register

Write the Verilog code, and test-bench for the shift register in Figure 2, and run the simulation to see how it works.

Expand your shift register to four bits and re-run the simulation.