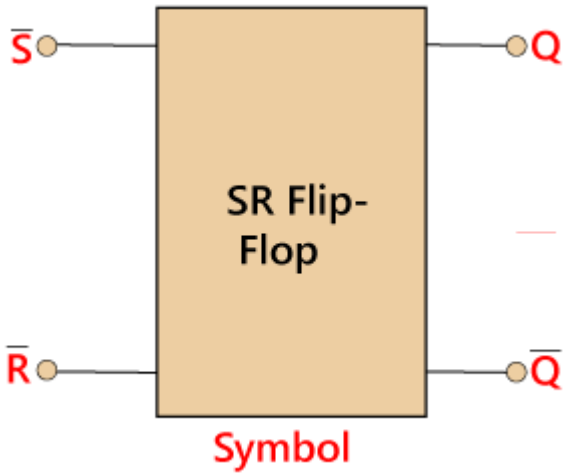
DAY 21

SR FLIPFLOP:

The SR flip flop is a 1-bit memory bistable device having two inputs, i.e., SET and RESET. The SET input 'S' set the device or produce the output 1, and the RESET input 'R' reset the device or produce the output 0. The SET and RESET inputs are labeled as S and R, respectively.

The SR flip flop stands for "Set-Reset" flip flop. The reset input is used to get back the flip flop to its original state from the current state with an output 'Q'. This output depends on the set and reset conditions, which is either at the logic level "0" or "1".



Symbol

The NAND gate SR flip flop is a basic flip flop which provides feedback from both of its outputs back to its opposing input. This circuit is used to store the single data bit in the memory circuit. So, the SR flip flop has a total of three inputs, i.e., 'S' and 'R', and current output 'Q'. This output 'Q' is related to the current history or state. The term "flip-flop" relates to the actual operation of the device, as it can be "flipped" to a logic set state or "flopped" back to the opposing logic reset state.

RTL CODE:

```
module SR_flipflop(
    input clk,
    input s,
    input r,
    output reg q,
    output qb
);

    always @(posedge clk) begin
        case({s, r})
            2'b00: q <= q;
            2'b01: q <= 1'b0;
            2'b10: q <= 1'b1;
            2'b11: q <= 1'bz;
        endcase
    end

    assign qb = ~q;

endmodule
```

TEST BENCH:

```
module testbench;
  reg clk,s,r;
  wire q,qb;
  SR_flipflop a1 (clk,s,r,q,qb);
  initial
    begin
      $dumpfile(".vcd");
      $dumpvars(1);
    end
  always
    begin

      clk = 0;
      #5 clk = 1;
      #5 clk = 0;
    end
  initial
    begin
      s = 0; r = 0;
      #10 s = 0; r = 1;
      #10 s = 1; r = 0;
      #10 s = 1; r = 1;
      #10 $finish;
```

```
    end
  always @(posedge clk) begin
    $display("Time %0t: s = %b, r = %b, q = %b, qb = %b", $time, s, r,
q, qb);
  end
endmodule
```