CS - 223

Lab - 4 Report

Section - 1

Mehmet Akif Şahin

22203673

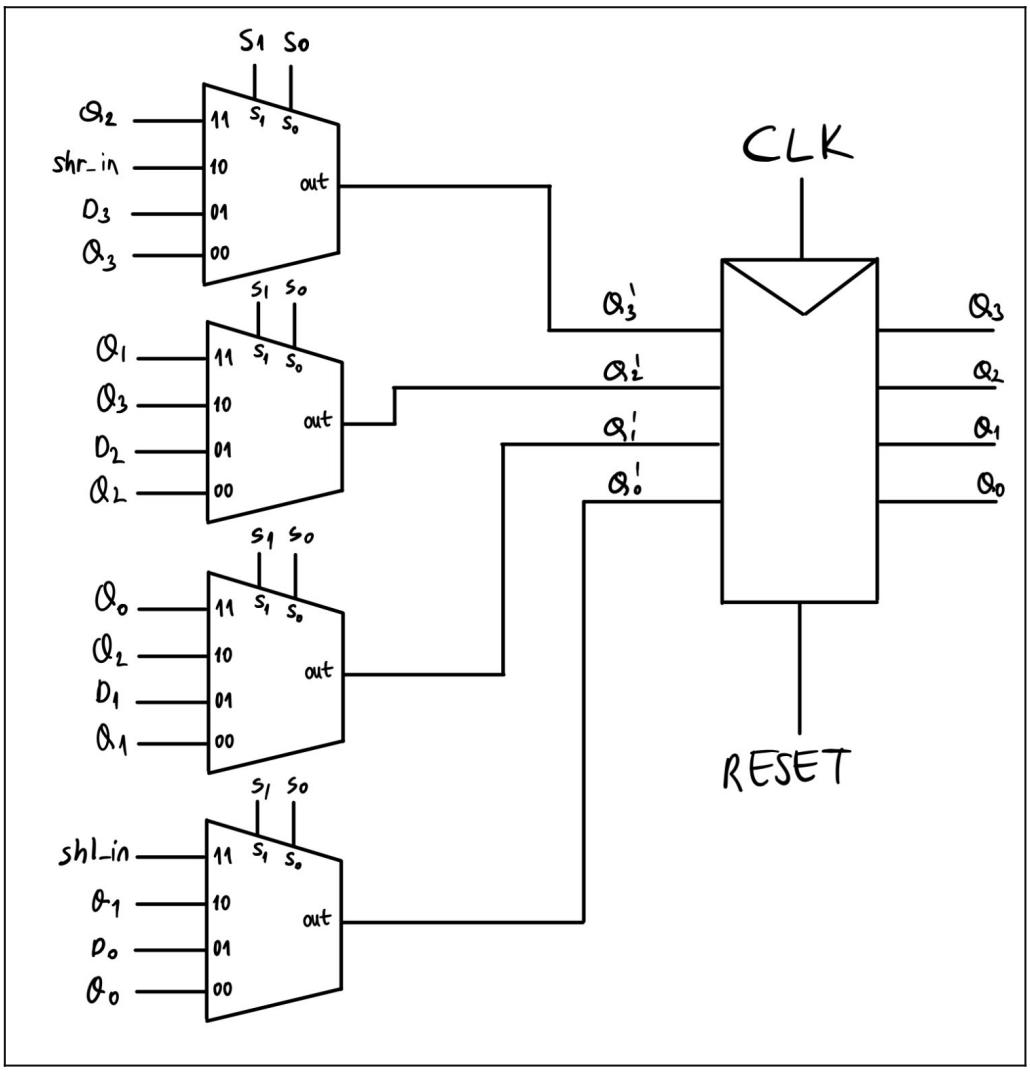
20.10.2023

(b) Write a SystemVerilog module for synchronously resettable D flip-flop.

```
module resettable_dflipflop
#(parameter input_width = 4)
(
   input clk,
   input [input_width-1:0] q_prime,
   input reset,
   output reg [input_width-1:0] q
   );

   always @ (posedge clk) begin
      if (reset) q <= 0;
      else q <= q_prime;
   end
endmodule
```

(c) Draw a circuit schematic (block diagram) for the internal design of the multifunction register by using 4:1 multiplexers and synchronously resettable D flip-flops.



(d) Write a Structural SystemVerilog module for the multifunction register you designed in part (c) and a testbench for it.

Module

```
module multifunction_register(
    input clk,
    input [3:0] d,
    input [1:0] s,
    input reset,
    input shr_in,
    input shl_in,
    output reg [3:0] q
    );

    wire [3:0] q_prime;

    mux_4to1 mux3({q[2], shr_in, d[3], q[3]}, s, q_prime[3]);
    mux_4to1 mux2({q[1], q[3], d[2], q[2]}, s, q_prime[2]);
    mux_4to1 mux1({q[0], q[2], d[1], q[1]}, s, q_prime[1]);
    mux_4to1 mux0({shl_in, q[1], d[0], q[0]}, s, q_prime[0]);

    resettable_dflipflop #(4) dflipflop (clk, q_prime, reset, q);

endmodule
```

```
module mux_4to1(
    input [3:0] d,
    input [1:0] s,
    output y
    );
    assign y = s[1] ? (s[0] ? d[3] : d[2]):(s[0] ? d[1] : d[0]);
endmodule
```

Testbench

```
module tb_mfregister(

    );

    reg clk;
    reg [3:0] q, d;
    reg [1:0] s;
    reg shr_in, shl_in;
    reg reset;

    multifunction_register uut (clk, d, s, reset, shr_in, shl_in, q);

    always begin
        clk = 1;
        #5;
        clk = 0;
```

```
    #5;
end

initial begin
    s = 2'b01;
    d = 4'b0000;
    shr_in = 0;
    shl_in = 0;
    reset = 0;
    #30;
    s = 2'b01;
    d = 4'b0110;
    shr_in = 0;
    shl_in = 0;
    reset = 0;
    #30;
    s = 2'b10;
    d = 4'b0000;
    shr_in = 1;
    shl_in = 0;
    reset = 0;
    #30;
    s = 2'b10;
    d = 4'b0000;
    shr_in = 0;
    shl_in = 0;
    reset = 0;
    #30;
    s = 2'b11;
    d = 4'b1010;
    shr_in = 0;
    shl_in = 1;
    reset = 1;
    #30;
    s = 2'b01;
    d = 4'b0011;
    shr_in = 0;
    shl_in = 1;
    reset = 0;
    #30;
    s = 2'b11;
    d = 4'b0000;
    shr_in = 0;
    shl_in = 1;
    reset = 0;
    #30;
    s = 2'b11;
    d = 4'b0000;
    shr_in = 0;
    shl_in = 0;
    reset = 0;
    #30;
```

```
        s = 2'b11;
        d = 4'b100;
        shr_in = 1;
        shl_in = 1;
        reset = 1;
        #30;
        s = 2'b01;
        d = 4'b1111;
        shr_in = 0;
        shl_in = 0;
        reset = 0;
        #30;
    end
endmodule
```