Chase Lotito

02/12/2024

ECE426

Dr. Chao Lu

HW4

Question 1:

q1.v

```
/*
Chase Lotito - SIUC Undergrad - SP2024
ECE426 - HW 4 - Q1
*/

module ff (
    reset_n, clk, data, select, d, data_out
);

// I/O
input reset_n, clk, data, select;
output data_out, d;

// Procedurals and continuous assignment
reg data_out;
wire d;

assign d = select ? data : data_out;

always @(posedge clk or reset_n) begin
    if (reset_n == 0)
        data_out = 0;
    else
        data_out <= d;
end

endmodule
```

tb.v

```
// Chase Lotito - HW4 Q1 Testbench

`include "q1.v"
`timescale 1us/1us
```

```
module tb ();

initial begin : GTKWAVE
    $dumpfile("tb.vcd");
    $dumpvars(0, tb);
end

// I/O
reg reset_n, clk, data, select;
wire d, data_out;

// CLOCK (start @ 0, 10us pulses, 5 cycles)
parameter cycles = 10;
initial begin : CLOCK
    clk = 0;
    repeat (cycles) begin
        #10 clk = ~clk;
    end
end

initial begin : stimmychecks
    reset_n = 1'b1; // no reset
    select = 1'b1; // send data to d
    data = 1'b1; // set data to 1

    #20 // prove posedge clk works
    select = 1'b1; // send data to d
    data = 1'b0; // set data to 1

    #5 // test async reset
    reset_n = 0;
    select = 0; // for d = data_out

    #5 // select=0, data=1 but it won't be registered!
    reset_n = 1;
    data = 1'b1;

    #5 // set data high and send to output
    select = 1'b1;


end

// Instantiate ff module
```

```
ff U1 (
    .reset_n(reset_n),
    .clk(clk),
    .data(data),
    .select(select),
    .d(d),
    .data_out(data_out)
);

endmodule
```

Question 2:

q2.v

```
/*
Chase Lotito - SIUC Undergrad - SP2024
ECE426 - HW 4 - Q2
*/

module ss (
    reset_n, clk, in0, in1, sel, d1, out
);

// I/O
input reset_n, clk, in0, in1, sel;
output d1, out;

// Procedural and continuous assignments
reg out;
wire d1; // redundant

// Next-state combinational logic
assign d1 = (in0 & ~sel) | (in1 & sel);

// Sequential logic for DFF
always @(posedge clk or reset_n) begin
    // out = 0 if reset_n == 0
    // out = d1 otherwise
    if (reset_n == 0) out <= 0;
    else out <= d1;
end

endmodule
```

tb.v

```
// Chase Lotito - HW4 Q2 Testbench

`include "q2.v"
`timescale 1us/1us

module tb ();

initial begin : GTKWAVE
    $dumpfile("tb.vcd");
    $dumpvars(0, tb);
end

// I/O
reg reset_n, clk, in0, in1, sel;
wire d1, out;

// CLOCK (start @ 0, 10us pulses, 5 cycles)
parameter cycles = 10;
initial begin : CLOCK
    clk = 0;
    repeat (cycles) begin
        #10 clk = ~clk;
    end
end

initial begin : stimmychecks
    in0 = 1'b0;
    in1 = 1'b1;
    reset_n = 1'b1;

    #10 // first posedge clk
    sel = 1'b1; // send in1 to d1

    #20
    sel = 1'b0; // send in0 to d1

    #20
    sel = 1'b1; // send in1 to d1

    #25 // middle of clk on
    reset_n = 1'b0; // reset!
```

```
end

// Module instantiation
ss U1 (
    .reset_n(reset_n),
    .clk(clk),
    .in0(in0),
    .in1(in1),
    .sel(sel),
    .d1(d1),
    .out(out)
);

endmodule
```