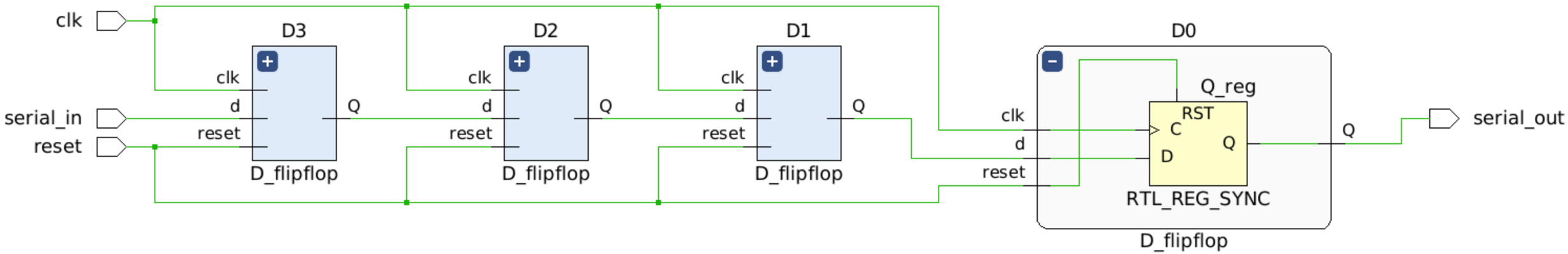
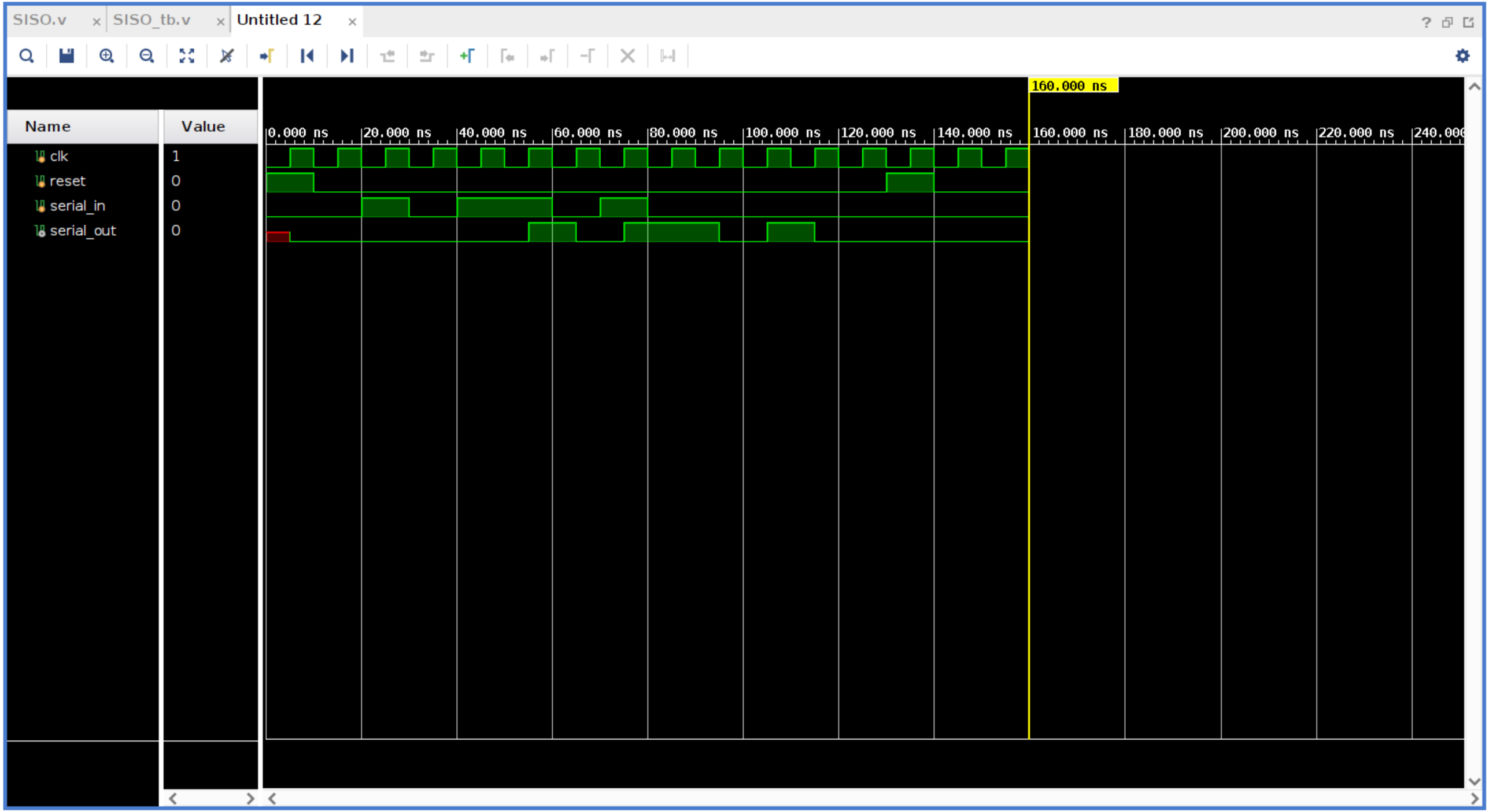
Project Summary | SISO.v | SISO_tb.v | Schematic

5 Cells 4 I/O Ports 11 Nets




SISO.v
SISO_tb.v
Untitled 12
Name
Value
clk
1
reset
0
serial_in
0
serial_out
0
0.000 ns
20.000 ns
40.000 ns
60.000 ns
80.000 ns
100.000 ns
120.000 ns
140.000 ns
160.000 ns
180.000 ns
200.000 ns
220.000 ns
240.000
160.000 ns

SISO.v * | SISO_tb.v | Untitled 12

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_055/project_1/project_1.srcs/sources_1/new/SISO.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Engineer: Anjan Prasad
// Create Date: 11/15/2024 10:16:08 AM
// Module Name: SISO
//////////////////////////////////////////////////////////////////////////////////

module SISO(
    input clk, reset, serial_in,
    output serial_out
    );

    wire q2, q1, q0;

    D_flipflop D3 (.d(serial_in), .clk(clk), .reset(reset), .Q(q2));
    D_flipflop D2 (.d(q2), .clk(clk), .reset(reset), .Q(q1));
    D_flipflop D1 (.d(q1), .clk(clk), .reset(reset), .Q(q0));
    D_flipflop D0 (.d(q0), .clk(clk), .reset(reset), .Q(serial_out));

endmodule

module D_flipflop(
    input clk, reset, d,
    output reg Q
    );

    always @(posedge clk) begin
        if (reset)
            Q <= 1'b0;
        else
            Q <= d;
    end

endmodule
```

SISO.v | SISO_tb.v | Untitled 12

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_055/project_1/project_1.srcs/sim_1/new/SISO_tb.v

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Engineer: Anjan Prasad
4  // Create Date: 11/15/2024 10:21:47 AM
5  // Module Name: SISO_tb
6  //////////////////////////////////////////////////////////////////////////////////
7  module SISO_tb;
8
9      reg clk, reset, serial_in;
10     wire serial_out;
11
12     SISO uut (
13         .clk(clk),
14         .reset(reset),
15         .serial_in(serial_in),
16         .serial_out(serial_out)
17     );
18     initial begin
19         clk = 0;
20         forever #5 clk = ~clk; // 10 ns period
21     end
22     initial begin
23         reset = 1;
24         serial_in = 0;
25         #10 reset = 0;
26         #10 serial_in = 1; // Input '1' -> expect serial_out = 0 after 4 cycles
27         #10 serial_in = 0;
28         #10 serial_in = 1;
29         #10 serial_in = 1;
30         #10 serial_in = 0;
31         #10 serial_in = 1;
32         #10 serial_in = 0;
33         #50;
34         reset = 1;
35         #10 reset = 0;
36         #20 $stop;
37     end
38 endmodule
39
```