In LAB: Implement 4-bit Universal Shift Register and verify through simulation

Code:

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 18:47:18 05/08/2018

// Design Name:

// Module Name: USR\_4bit

// Project Name:

// Target Devices:

// Tool versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module USR\_4bit(

input CLRb,

input S1,

input S0,

input CLK,

input SDL,

input SDR,

output [3:0] D,

output [3:0] Q

);

reg[3:0] int\_Q;

always @(posedge CLK or negedge CLRb) begin

if (CLRb == 1'b0) begin

int\_Q <= 4'b0000;

end

else begin

if(S1 == S0) begin

int\_Q = (S1 == 1'b1) ? D : Q;

end

else begin

if(S1 == 1'b1) begin

int\_Q[3:1] = Q[2:0];

if(SDL == 1'b0) begin

int\_Q[0] = 1'b0;

end

else begin

int\_Q[0] = 1'b1;

end

end

else begin

int\_Q[2:0] = Q[3:1];

if(SDR == 1'b0) begin

int\_Q[3] = 1'b0;

end

else begin

int\_Q[3] = 1'b1;

end

end

end

end

end

assign Q = int\_Q;

endmodule

Test Code:

`timescale 1ns / 1ps

////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 18:48:39 05/08/2018

// Design Name: USR\_4bit

// Module Name: /csehome/pistolstar1797/USR\_4bit/USR\_4bit\_test.v

// Project Name: USR\_4bit

// Target Device:

// Tool versions:

// Description:

//

// Verilog Test Fixture created by ISE for module: USR\_4bit

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

////////////////////////////////////////////////////////////////////////////////

module USR\_4bit\_test;

// Inputs

reg CLRb;

reg S1;

reg S0;

reg SDL;

reg SDR;

wire CLK;

reg ENA;

// Outputs

reg [3:0] D;

wire [3:0] Q;

// Instantiate the Unit Under Test (UUT)

USR\_4bit uut1 (

.CLRb(CLRb),

.S1(S1),

.S0(S0),

.SDL(SDL),

.SDR(SDR),

.CLK(CLK),

.D(D),

.Q(Q)

);

CLKGEN uut2 (

.ENA(ENA),

.CLK(CLK)

);

initial begin

// Initialize Inputs

ENA = 1'b0;

#15;

ENA = 1'b1;

#15;

CLRb = 1'b0;

S1 = 1'b0;

S0 = 1'b0;

SDL = 1'b0;

SDR = 1'b0;

D = 4'b1111;

#15;

CLRb = 1'b1;

D = 4'b1010;

#15;

S1 = 1'b1;

S0 = 1'b1;

#15;

S1 = 1'b1;

S0 = 1'b0;

#15;

SDL = 1;

#15;

S1 = 1'b0;

S0 = 1'b1;

#15;

SDR = 1;

#15;

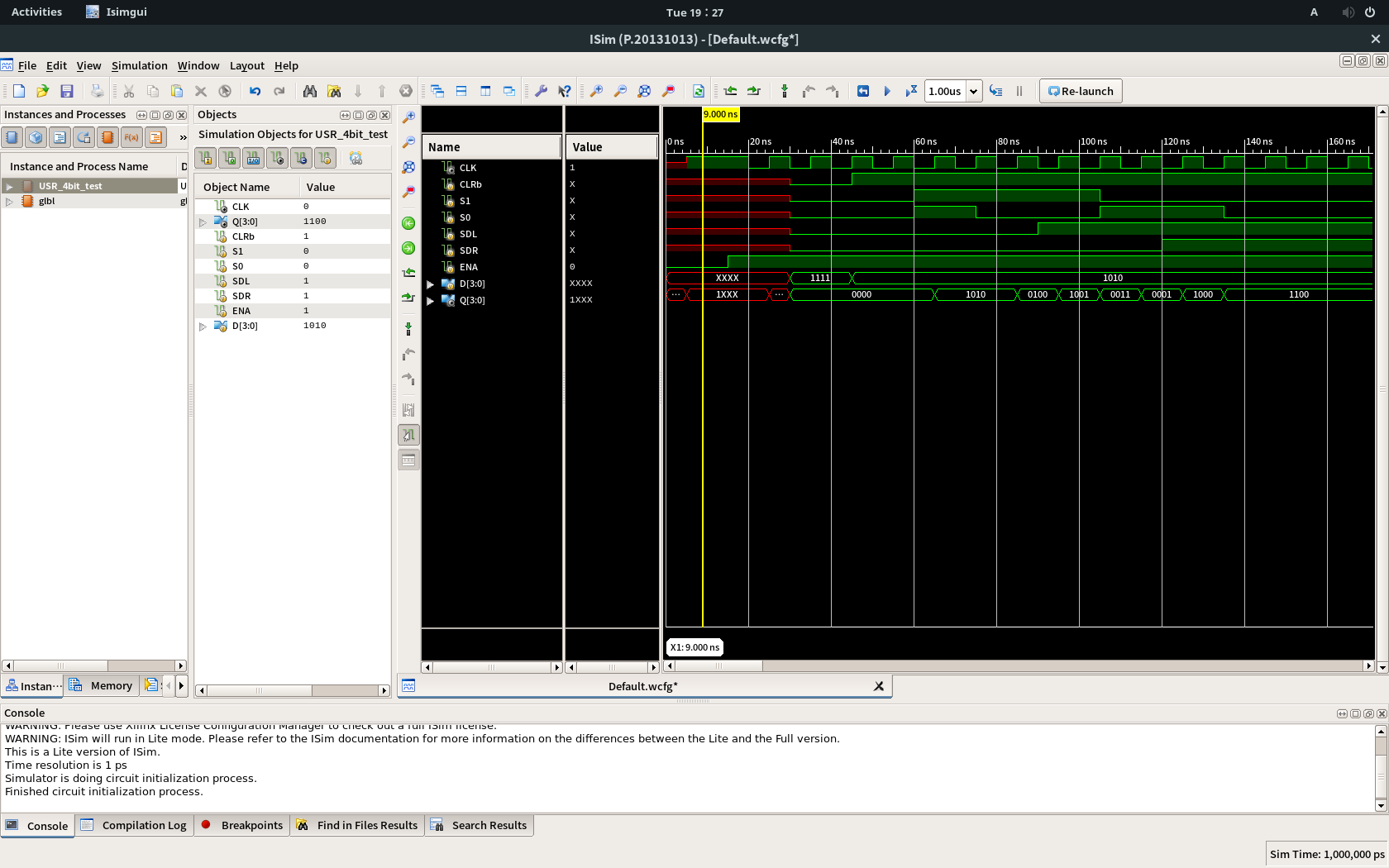
S1 = 1'b0;

S0 = 1'b0;

end

endmodule

실행 결과:



Homework: Implement & Simulate 8-bit Universal Shift Register

Code:

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 19:16:55 05/08/2018

// Design Name:

// Module Name: USR\_8bit

// Project Name:

// Target Devices:

// Tool versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module USR\_8bit(

input CLRb,

input S1,

input S0,

input CLK,

input SDL,

input SDR,

output [7:0] D,

output [7:0] Q

);

reg[7:0] int\_Q;

always @(posedge CLK or negedge CLRb) begin

if (CLRb == 1'b0) begin

int\_Q <= 8'b00000000;

end

else begin

if(S1 == S0) begin

int\_Q = (S1 == 1'b1) ? D : Q;

end

else begin

if(S1 == 1'b1) begin

int\_Q[7:1] = Q[6:0];

if(SDL == 1'b0) begin

int\_Q[0] = 1'b0;

end

else begin

int\_Q[0] = 1'b1;

end

end

else begin

int\_Q[6:0] = Q[7:1];

if(SDR == 1'b0) begin

int\_Q[7] = 1'b0;

end

else begin

int\_Q[7] = 1'b1;

end

end

end

end

end

assign Q = int\_Q;

endmodule

Test Code:

`timescale 1ns / 1ps

////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 19:18:34 05/08/2018

// Design Name: USR\_8bit

// Module Name: /csehome/pistolstar1797/USR\_4bit/USR\_8bit\_test.v

// Project Name: USR\_4bit

// Target Device:

// Tool versions:

// Description:

//

// Verilog Test Fixture created by ISE for module: USR\_8bit

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

////////////////////////////////////////////////////////////////////////////////

module USR\_8bit\_test;

// Inputs

reg CLRb;

reg S1;

reg S0;

wire CLK;

reg SDL;

reg SDR;

reg ENA;

// Outputs

reg [7:0] D;

wire [7:0] Q;

// Instantiate the Unit Under Test (UUT)

USR\_8bit uut1 (

.CLRb(CLRb),

.S1(S1),

.S0(S0),

.CLK(CLK),

.SDL(SDL),

.SDR(SDR),

.D(D),

.Q(Q)

);

CLKGEN uut2 (

.CLK(CLK),

.ENA(ENA)

);

initial begin

// Initialize Inputs

ENA = 1'b0;

#15;

ENA = 1'b1;

#15;

CLRb = 1'b0;

S1 = 1'b0;

S0 = 1'b0;

SDL = 1'b0;

SDR = 1'b0;

D = 8'b11111111;

#15;

CLRb = 1'b1;

D = 8'b10101010;

#15;

S1 = 1'b1;

S0 = 1'b1;

#15;

S1 = 1'b1;

S0 = 1'b0;

#15;

SDL = 1;

#15;

S1 = 1'b0;

S0 = 1'b1;

#15;

SDR = 1;

#15;

S1 = 1'b0;

S0 = 1'b0;

end

endmodule

실행 결과:

