**Experiment 5: Flipflops**

Jyothiradithyan k

S3 CSE, 34.

Date: 18/06/21

**AIM**

(a) Development of Verilog modules for SR, JK, T and D flip flops.

**SR flipflop**

**VERILOG CODE**

**Behavioural level**

module SR\_ff(

input S,R,clk,clr,

output reg q,qbar);

//intial

always@(posedge clk begin)

begin

if (clr)

begin

q<=1'b0; //nonblocking

qbar<=1'b1;

end

else if((S==1)&&(R==0))

begin

q<=1'b1; //nonblocking

qbar<=1'b0;

end

else if((S==0)&&(R==1))

begin

q<=1'b0; //nonblocking

qbar<=1'b1;

end

else if((S==0)&&(R==0))

begin

q<=q; //nonblocking

qbar<=qbar;

end

else if((S==1)&&(R==1))

begin

q<=1'bx; //nonblocking

qbar<=1'bx;

end

end

endmodule

**Testbench**

`timescale 1ns/1ps

module SR\_FFtest;

reg t\_S,t\_R,t\_clk,t\_clr;

wire t\_q,t\_qbar:

SR\_ff uut(.S(t\_S),.R(t\_R),.clk(t\_clk),.clr(t\_clr),.q(t\_q),.qbar(t\_qbar));

initial

begin

$dumpvars(1,SR\_FFtest);

t\_clr=1;

t clk=0;

#10;

t clr=0;

t\_S=1;

t\_R=0;

#10;

t clr=0;

t\_S=0;

t\_R=1;

#10;

t clr=0;

t\_S=0;

t\_R=0;

#10;

t clr=0;

t\_S=1;

t\_R=1;

#10;

$stop;

end

always

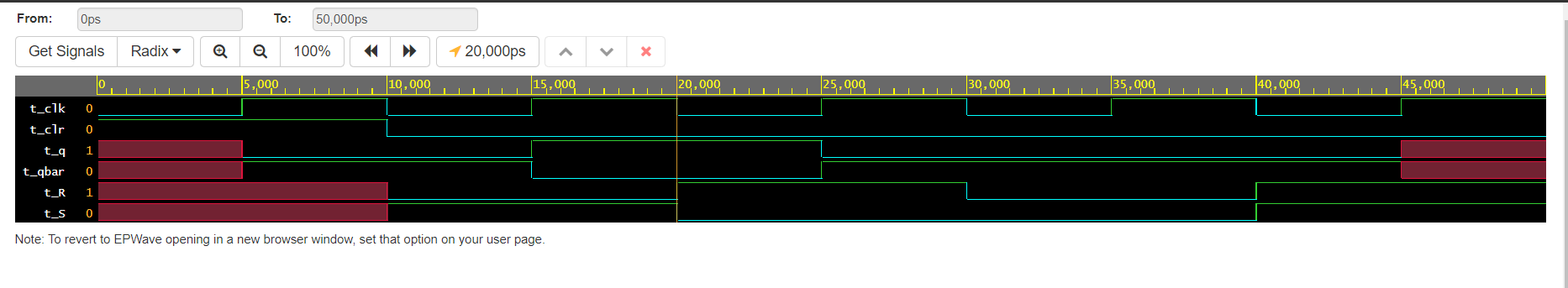
begin

#5 t\_clk = ~t\_clk;

end

endmodule

**Output**

****

**JK FLIPFLOP**

**VERILOG CODE**

**Behavioural level**

module JK\_ff(

input J,K,clk,clr,

output reg q,qbar);

//intial

always@(posedge clk)

begin

if (clr)

begin

q<=1'b0; //nonblocking

qbar<=1'b1;

end

else if((J==1)&&(K==0))

begin

q<=1'b1; //nonblocking

qbar<=1'b0;

end

else if((J==0)&&(K==1))

begin

q<=1'b0; //nonblocking

qbar<=1'b1;

end

else if((J==0)&&(K==0))

begin

q<=q; //nonblocking

qbar<=qbar;

end

else if((J==1)&&(K==1))

begin

q<= ~q; //nonblocking

qbar<= ~qbar;

end

end

endmodule

**Testbench**

`timescale 1ns/1ps

module JK\_FFtest;

reg t\_J,t\_K,t\_clk,t\_clr;

wire t\_q,t\_qbar;

JK\_ff uut(.J(t\_J),.K(t\_K),.clk(t\_clk),.clr(t\_clr),.q(t\_q),.qbar(t\_qbar));

initial

begin

$dumpvars(1,JK\_FFtest);

t\_clr=1;

t\_clk=0;

#10;

t\_clr=0;

t\_J=1;

t\_K=0;

#10;

t\_clr=0;

t\_J=0;

t\_K=1;

#10;

t\_clr=0;

t\_J=0;

t\_K=0;

#10;

t\_clr=0;

t\_J=1;

t\_K=1;

#10;

$stop;

end

always

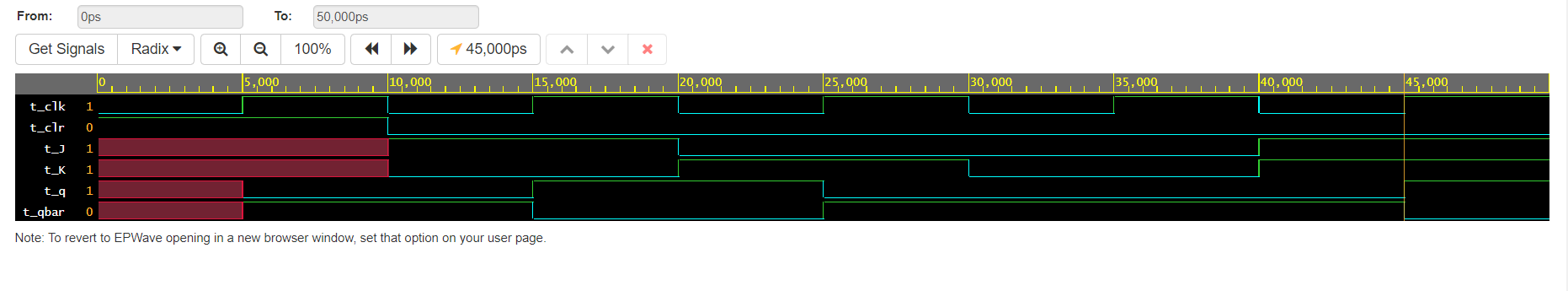
begin

#5 t\_clk = ~t\_clk;

end

endmodule

**Output**

****

**T FLIPFLOP**

**VERILOG CODE**

**Behavioural level**

module T\_ff(

input T,clk,clr,

output reg q,qbar);

//intial

always@(posedge clk)

begin

if (clr)

begin

q<=1'b0; //nonblocking

qbar<=1'b1;

end

else if(T==0)

begin

q<=q; //nonblocking

qbar<=qbar;

end

else if(T==1)

begin

q<= ~q; //nonblocking

qbar<= ~qbar;

end

end

endmodule

**Testbench**

`timescale 1ns/1ps

module T\_FFtest;

reg t\_T,t\_clk,t\_clr;

wire t\_q,t\_qbar;

T\_ff uut(.T(t\_T),.clk(t\_clk),.clr(t\_clr),.q(t\_q),.qbar(t\_qbar));

initial

begin

$dumpvars(1,T\_FFtest);

t\_clr=1;

t\_clk=0;

#10;

t\_clr=0;

t\_T=1;

#10;

t\_clr=0;

t\_T=0;

#10;

$stop;

end

always

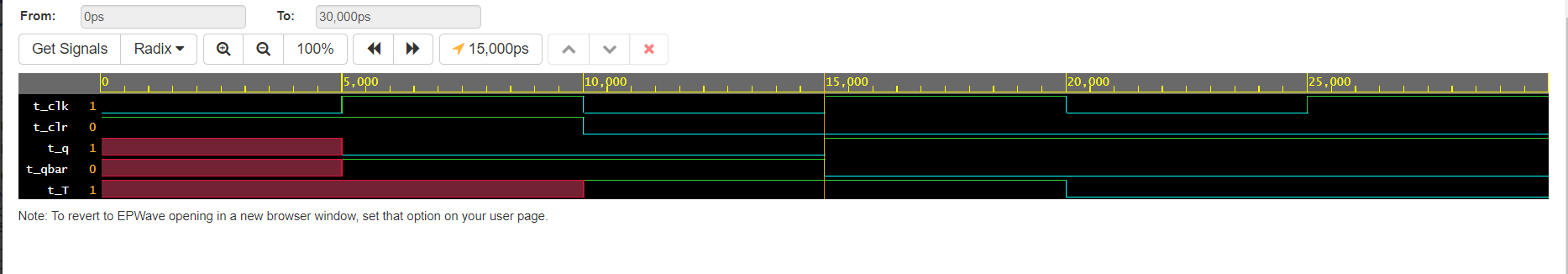
begin

#5 t\_clk = ~t\_clk;

end

endmodule

**Output**

****

**D FLIPFLOP**

**VERILOG CODE**

**Behavioural level**

module D\_ff(

input D,clk,clr,

output reg q,qbar);

//intial

always@(posedge clk)

begin

if (clr)

begin

q<=1'b0; //nonblocking

qbar<=1'b1;

end

else if(D==0)

begin

q<=1'b0; //nonblocking

qbar<=1'b1;

end

else if(D==1)

begin

q<=1'b1; //nonblocking

qbar<=1'b0;

end

end

endmodule

**Testbench**

`timescale 1ns/1ps

module D\_FFtest;

reg t\_D,t\_clk,t\_clr;

wire t\_q,t\_qbar;

D\_ff uut(.D(t\_D),.clk(t\_clk),.clr(t\_clr),.q(t\_q),.qbar(t\_qbar));

initial

begin

$dumpvars(1,D\_FFtest);

t\_clr=1;

t\_clk=0;

#10;

t\_clr=0;

t\_D=1;

#10;

t\_clr=0;

t\_D=0;

#10;

$stop;

end

always

begin

#5 t\_clk = ~t\_clk;

end

endmodule

**Output**

