Noah Kessler

Lab 2

Part 1

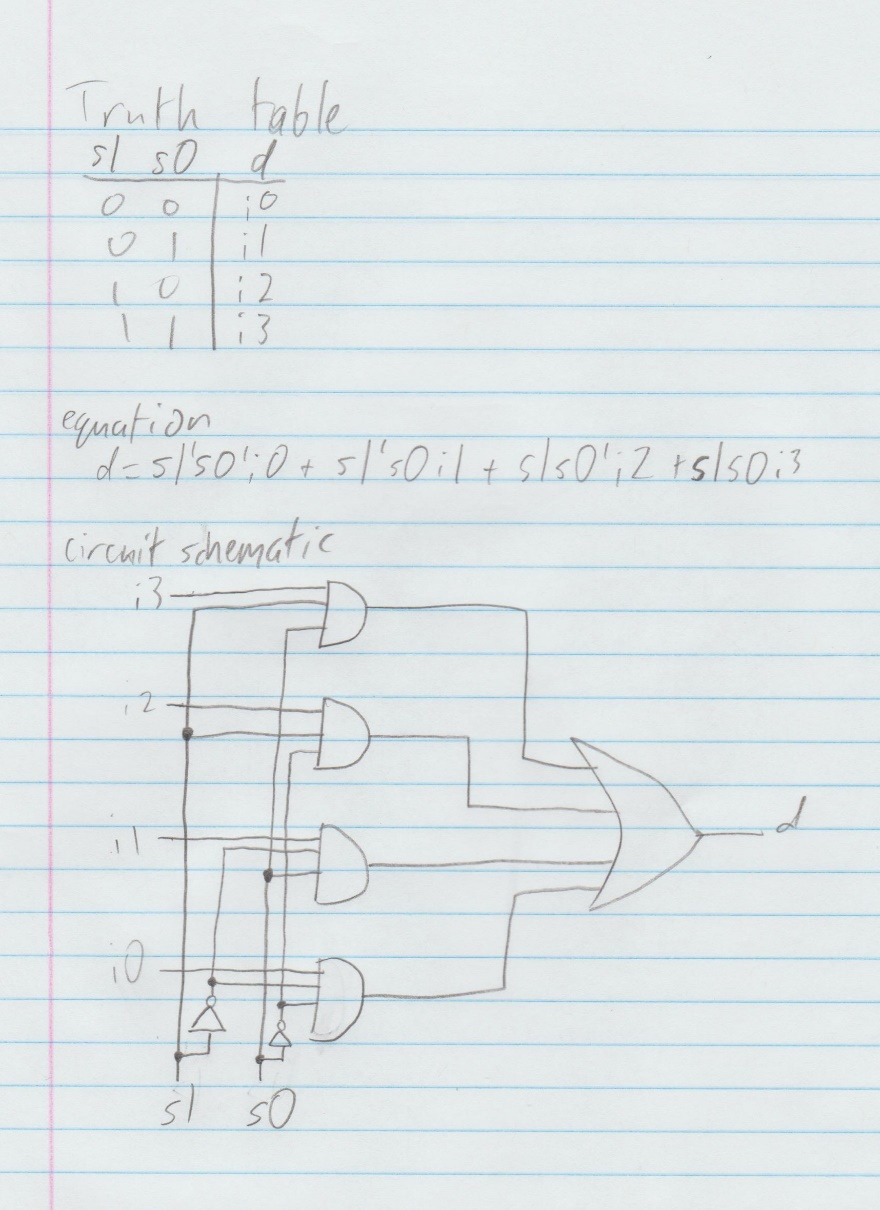
Structural



Behavioral



Part 2



Structural

module Mux\_structural(

input i0,

input i1,

input i2,

input i3,

input s0,

input s1,

output d

);

wire s0\_not, s1\_not;

wire d0, d1, d2, d3;

not n0 (s0\_not, s0);

not n1 (s1\_not, s1);

and g0 (d0, i0, s1\_not, s0\_not);

and g1 (d1, i1, s1\_not, s0);

and g2 (d2, i2, s1, s0\_not);

and g3 (d3, i3, s1, s0);

or g4 (d, d0, d1, d2, d3);

endmodule

module tb\_Mux\_Structural;

reg i0;

reg i1;

reg i2;

reg i3;

reg s0;

reg s1;

wire d;

Mux\_structural uut (

.i0(i0),

.i1(i1),

.i2(i2),

.i3(i3),

.s0(s0),

.s1(s1),

.d(d)

);

initial begin

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

s1 = 0;

s0 = 0;

#50;

s1 = 0;

s0 = 0;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC01");

if (d != 1'b0) $display ("Result is wrong");

s1 = 0;

s0 = 0;

i0 = 1;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC02");

if (d != 1'b0) $display ("Result is wrong");

s1 = 0;

s0 = 1;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC03");

if (d != 1'b0) $display ("Result is wrong");

s1 = 0;

s0 = 1;

i0 = 0;

i1 = 1;

i2 = 0;

i3 = 0;

#50;

$display ("TC04");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 0;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC05");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 0;

i0 = 0;

i1 = 0;

i2 = 1;

i3 = 0;

#50;

$display ("TC06");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 1;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC07");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 1;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 1;

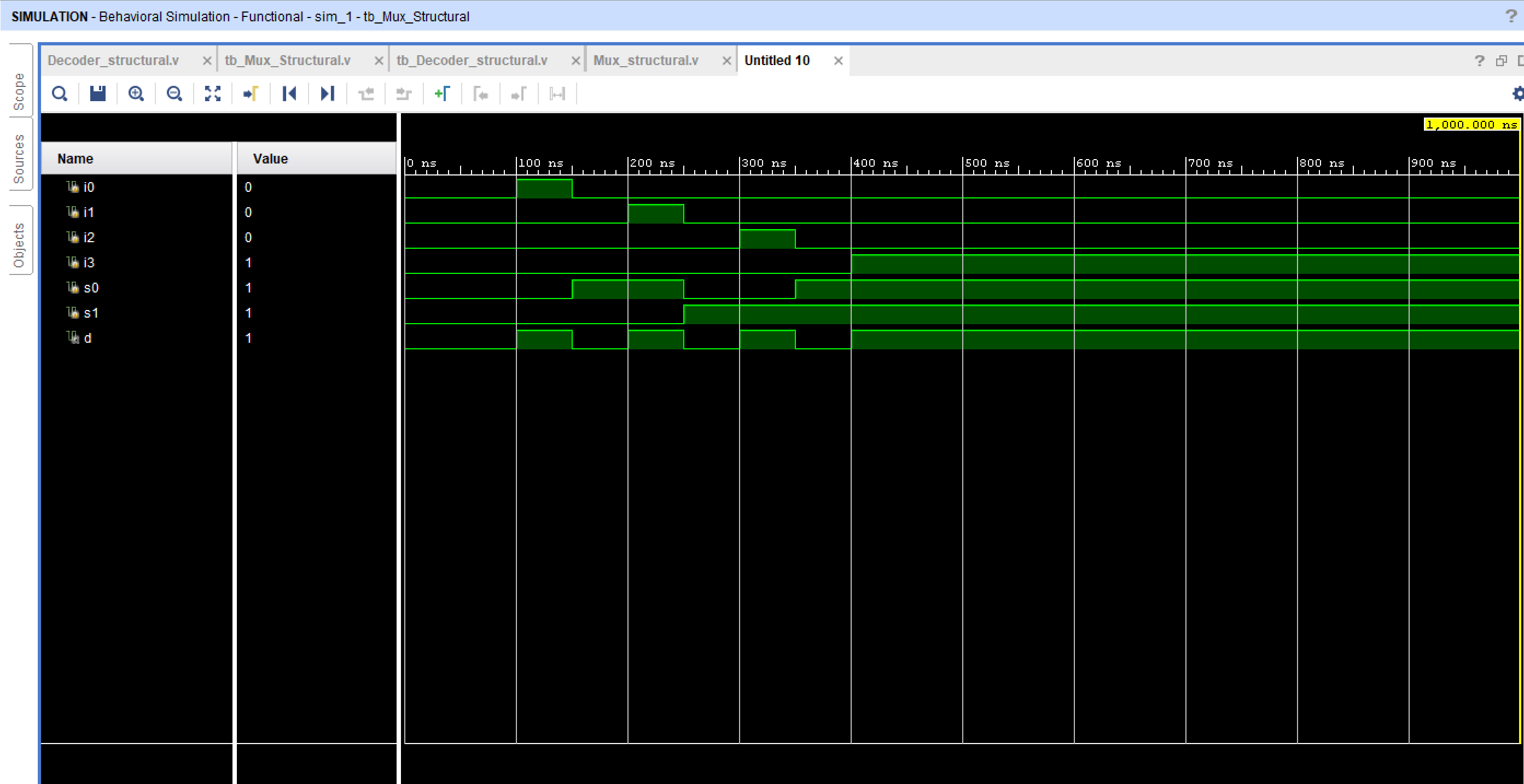
#50;

$display ("TC08");

if (d != 1'b0) $display ("Result is wrong");

end

endmodule



Behavioral

module Mux\_behavioral(

input i0,

input i1,

input i2,

input i3,

input s0,

input s1,

output reg d

);

always @(i0, i1, i2, i3, s0, s1)

begin

d=1'b0;

case({s1, s0})

2'b00 : d=i0;

2'b01 : d=i1;

2'b10 : d=i2;

2'b11 : d=i3;

default : begin

d=1'b0;

end

endcase

end

endmodule

module tb\_Mux\_behavioral;

reg i0;

reg i1;

reg i2;

reg i3;

reg s0;

reg s1;

wire d;

Mux\_behavioral uut (

.i0(i0),

.i1(i1),

.i2(i2),

.i3(i3),

.s0(s0),

.s1(s1),

.d(d)

);

initial begin

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

s1 = 0;

s0 = 0;

#50;

s1 = 0;

s0 = 0;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC01");

if (d != 1'b0) $display ("Result is wrong");

s1 = 0;

s0 = 0;

i0 = 1;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC02");

if (d != 1'b0) $display ("Result is wrong");

s1 = 0;

s0 = 1;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC03");

if (d != 1'b0) $display ("Result is wrong");

s1 = 0;

s0 = 1;

i0 = 0;

i1 = 1;

i2 = 0;

i3 = 0;

#50;

$display ("TC04");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 0;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC05");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 0;

i0 = 0;

i1 = 0;

i2 = 1;

i3 = 0;

#50;

$display ("TC06");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 1;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 0;

#50;

$display ("TC07");

if (d != 1'b0) $display ("Result is wrong");

s1 = 1;

s0 = 1;

i0 = 0;

i1 = 0;

i2 = 0;

i3 = 1;

#50;

$display ("TC08");

if (d != 1'b0) $display ("Result is wrong");

end

endmodule

