**Vending machine with change machine**

Verilog code

module vending\_machine\_18105070(

input clk,

input rst,

input [1:0]in, // 01 = 5 rs, 10 = 10 rs

output reg out,

output reg[1:0] change

);

parameter s0 = 2'b00;

parameter s1 = 2'b01;

parameter s2 = 2'b10;

reg[1:0] c\_state,n\_state;

always@ (posedge clk)

begin

if(rst == 1)

begin

c\_state = 0;

n\_state = 0;

change = 2'b00;

end

else

c\_state = n\_state;

case(c\_state)

s0: //state 0 : 0 rs

if(in == 0)

begin

n\_state = s0;

out = 0;

change = 2'b00;

end

else if(in == 2'b01)

begin

n\_state = s1;

out = 0;

change = 2'b00;

end

else if(in == 2'b10)

begin

n\_state = s2;

out = 0;

change = 2'b00;

end

s1: //state 1 : 5 rs

if(in == 0)

begin

n\_state = s0;

out = 0;

change = 2'b01; //change returned 5 rs

end

else if(in == 2'b01)

begin

n\_state = s2;

out = 0;

change = 2'b00;

end

else if(in == 2'b10)

begin

n\_state = s0;

out = 1;

change = 2'b00;

end

s2: //state 2 : 10 rs

if(in == 0)

begin

n\_state = s0;

out = 0;

change = 2'b10;

end

else if(in == 2'b01)

begin

n\_state = s0;

out = 1;

change = 2'b00;

end

else if(in == 2'b10)

begin

n\_state = s0;

out = 1;

change = 2'b01; //change returned 5 rs and 1 bottle

end

endcase

end

endmodule

Testbench

module vending\_machine\_tb;

//inputs

reg clk;

reg[1:0] in;

reg rst;

//output

wire out;

wire[1:0] change;

vending\_machine\_18105070 uut(

.clk(clk),

.rst(rst),

.in(in),

.out(out),

.change(change)

);

initial begin

//initialise inputs

$dumpfile("vending\_machine\_18105070.vcd");

$dumpvars(0,vending\_machine\_tb);

rst = 1;

clk = 0;

#6 rst = 0;

in = 2;

#19 in = 2;

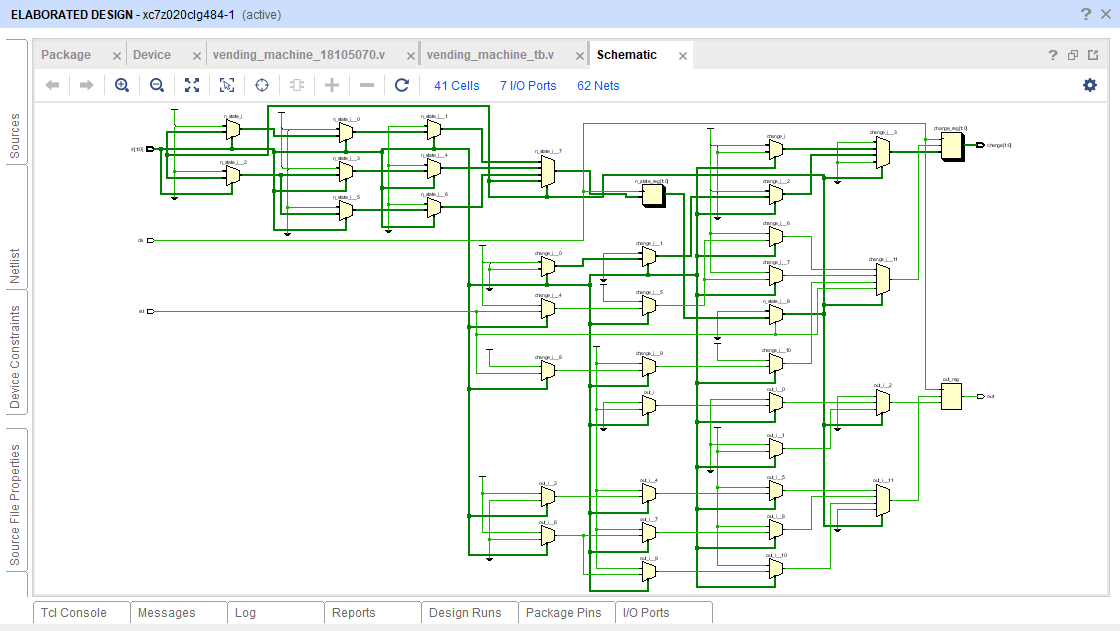
#25 $finish;

end

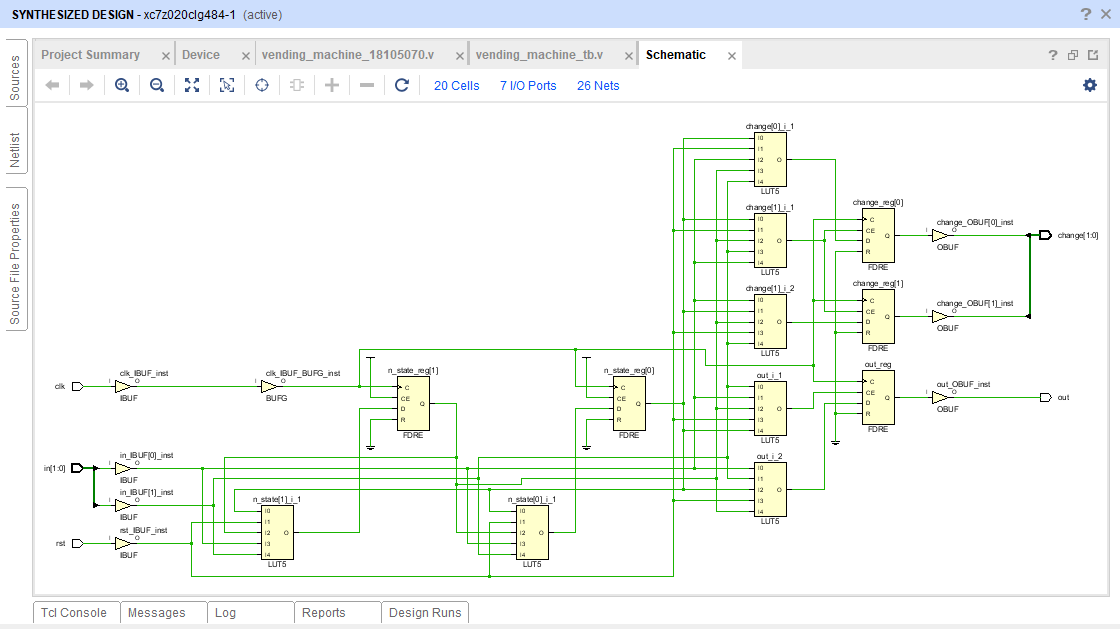
always #5 clk = ~clk;

endmodule

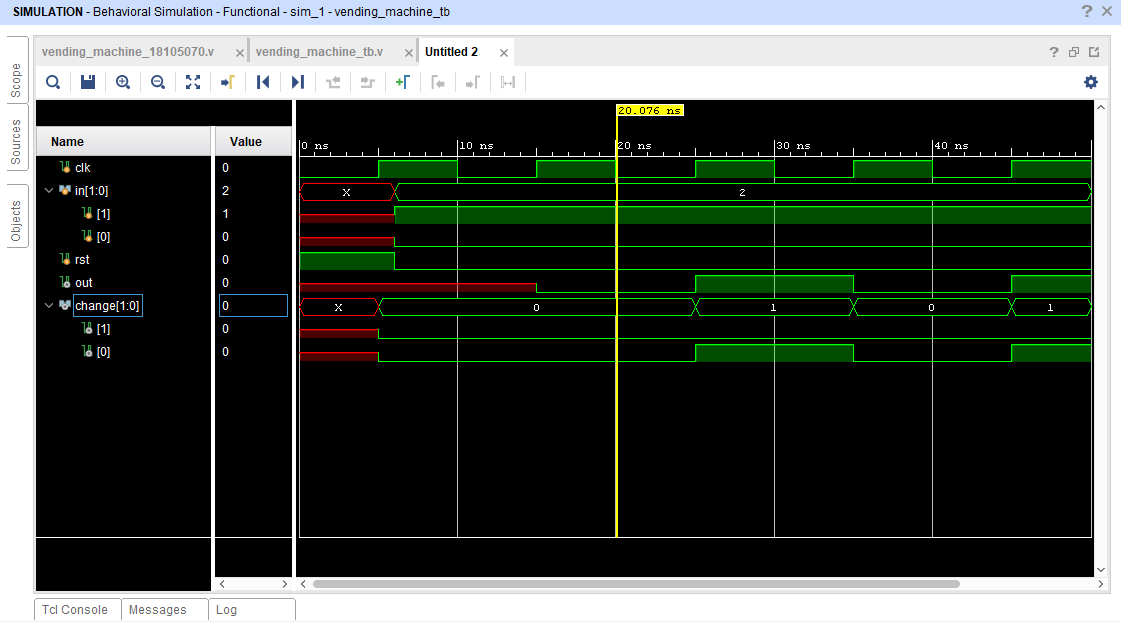
schematic



**Synthesis design/Implemented design**



**Behavioural simulation**



**THE END**