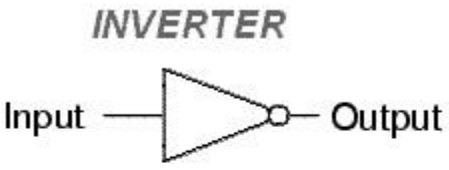
Maria Moșneag
343C1

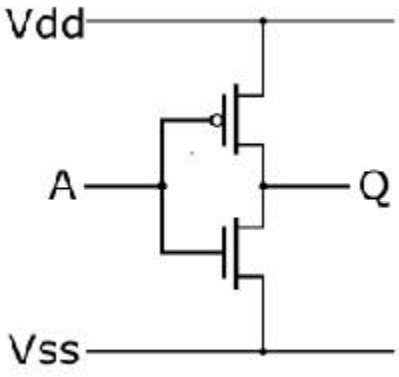
Referat VLSI

1. Inversor

a. Schema circuitului

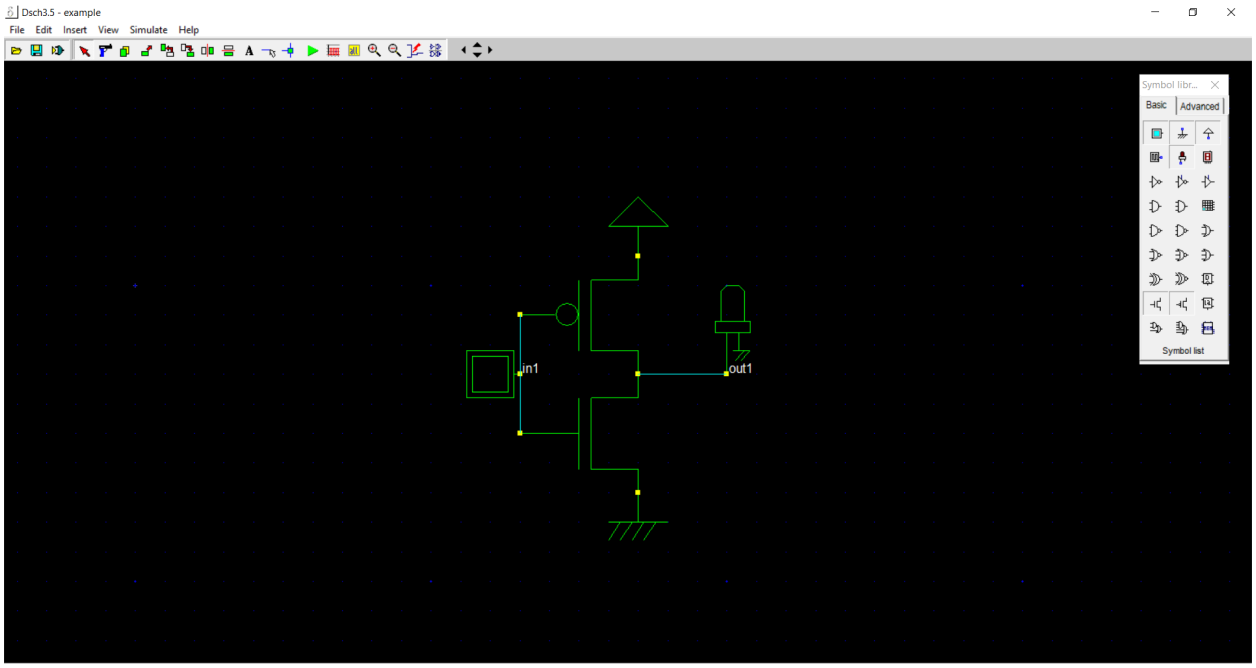


Input	Output
1	0
0	1

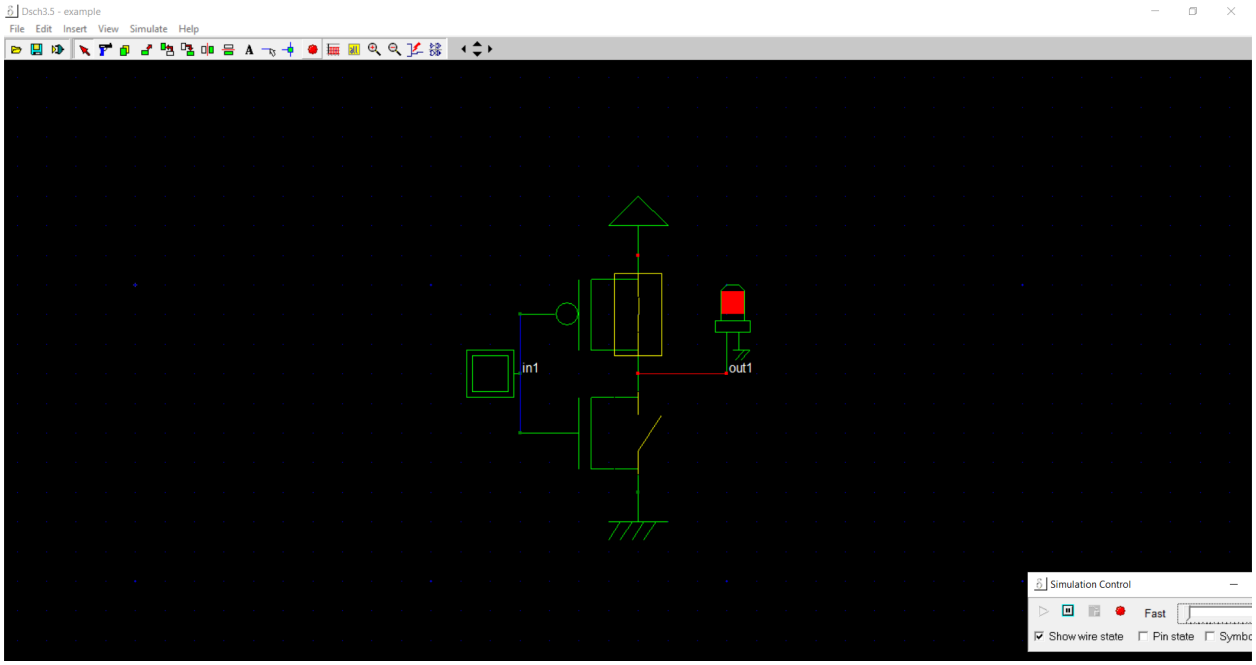


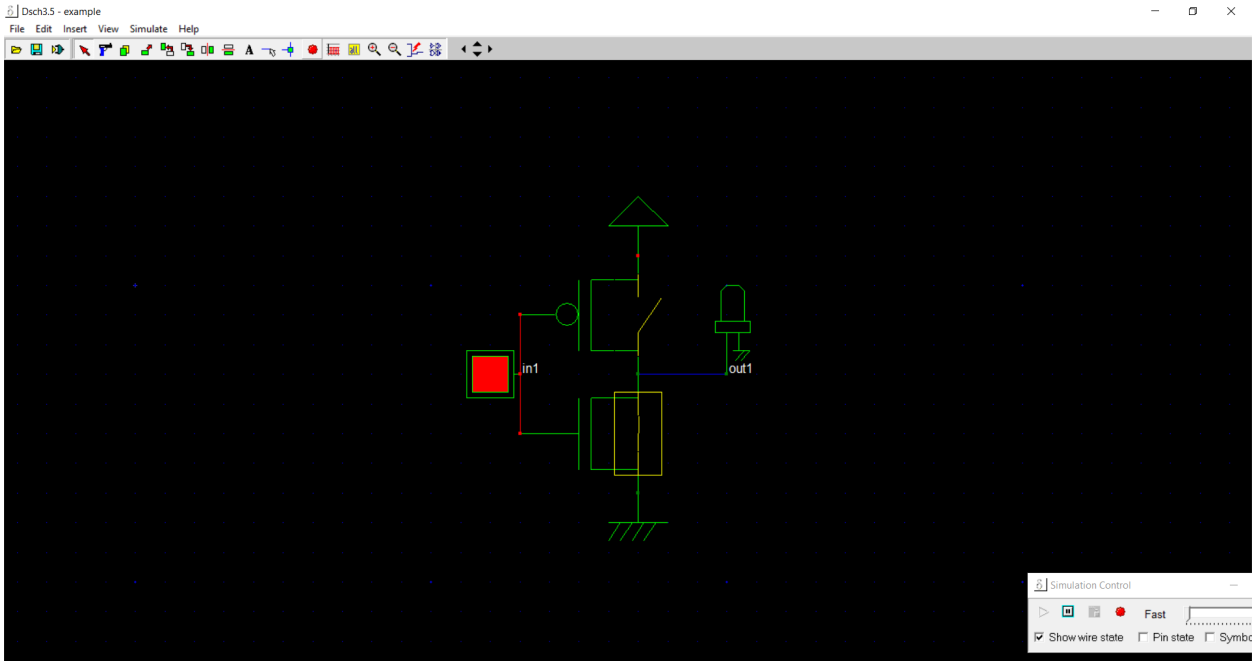
Sursa: https://vlab.amrita.edu/index.php?sub=59&brch=165&sim=901&cnt=1

b. Implementarea în DSCH



c. Simularea în DSCH

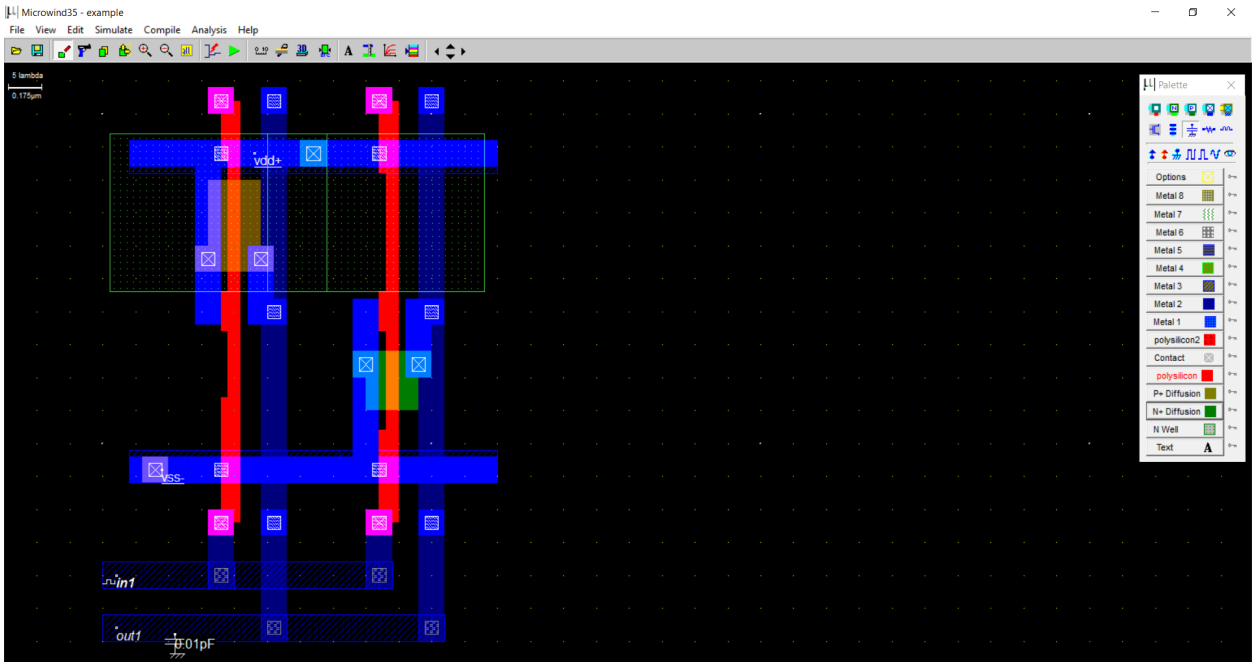




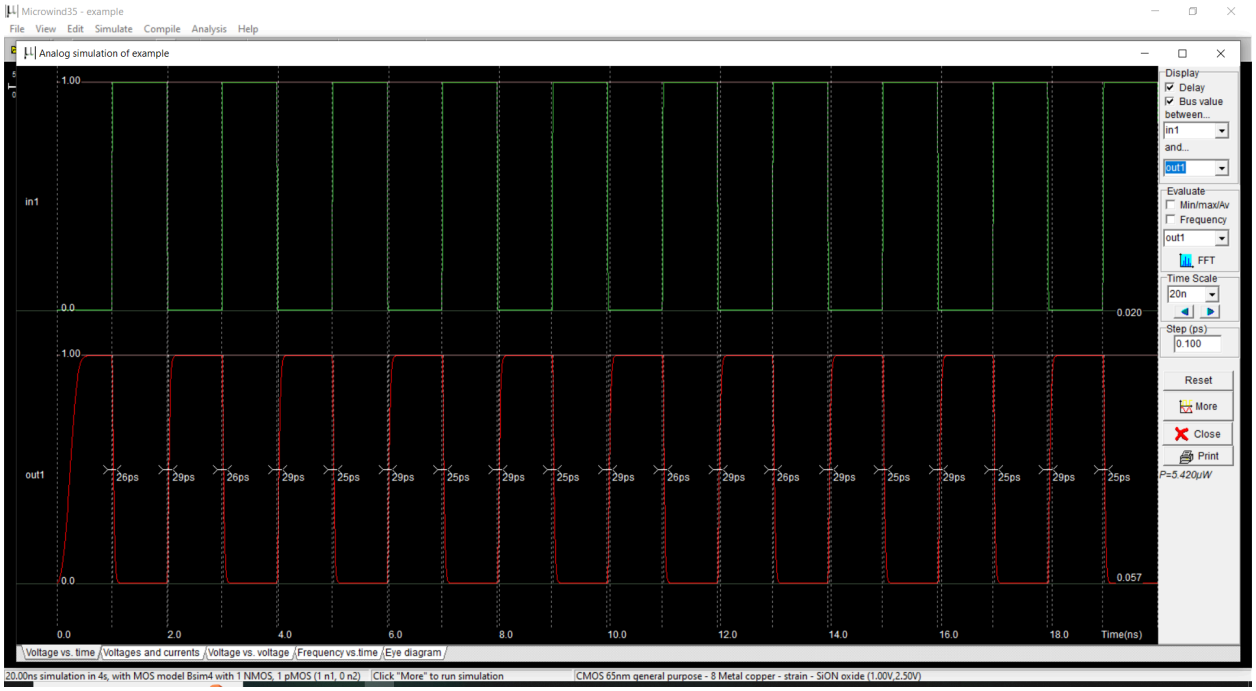
d. Codul Verilog

```
module inversor( in1,out1);
 input in1;
 output out1;
 wire ;
 pmos #(2) pmos_1(out1,vdd,in1); // 0.5u 0.07u
 nmos #(2) nmos_2(out1,vss,in1); // 0.3u 0.07u
endmodule

// Simulation parameters in Verilog Format
always
#200 in1=~in1;

// Simulation parameters
// in1 CLK 1 1
```

e. Implementarea în Microwind



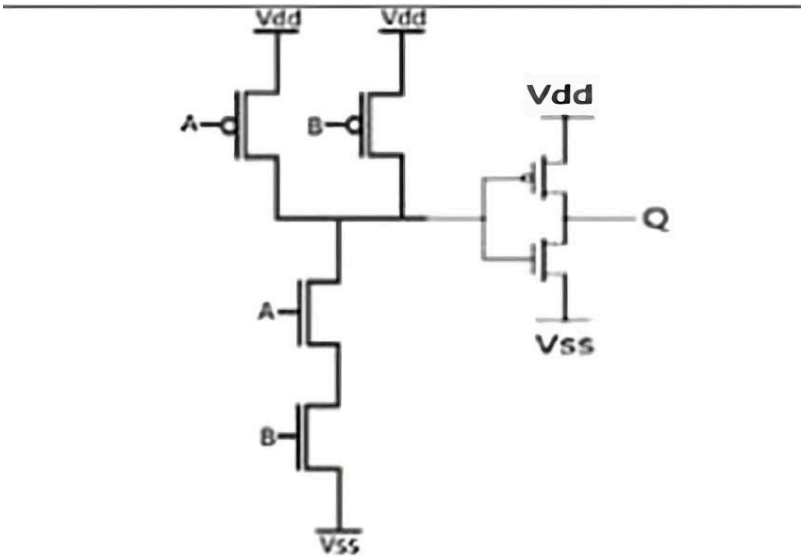
f. Simulare Microwind



2. AND

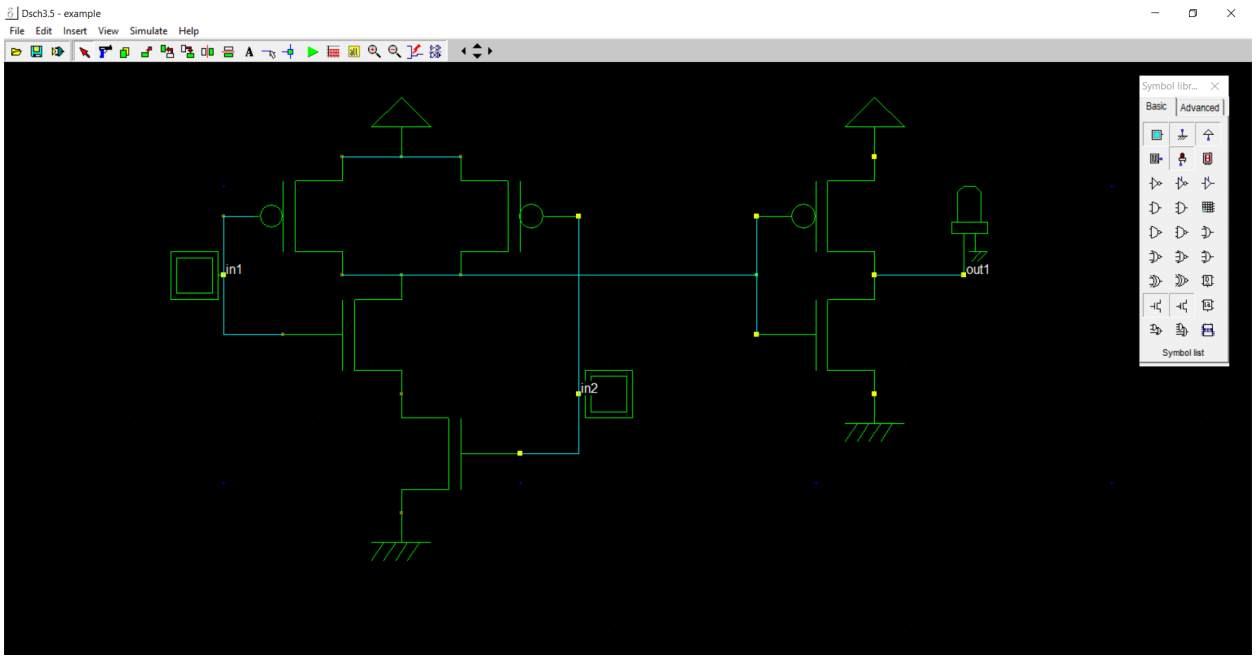
a. Schema circuitului

b.

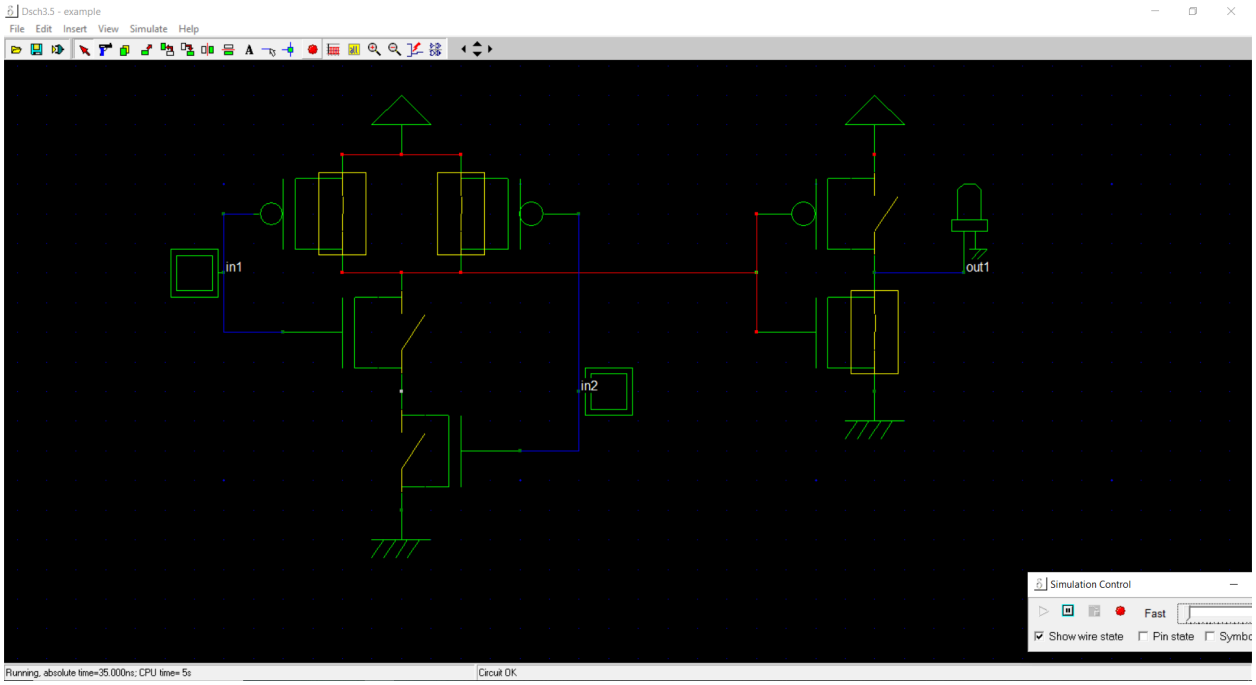


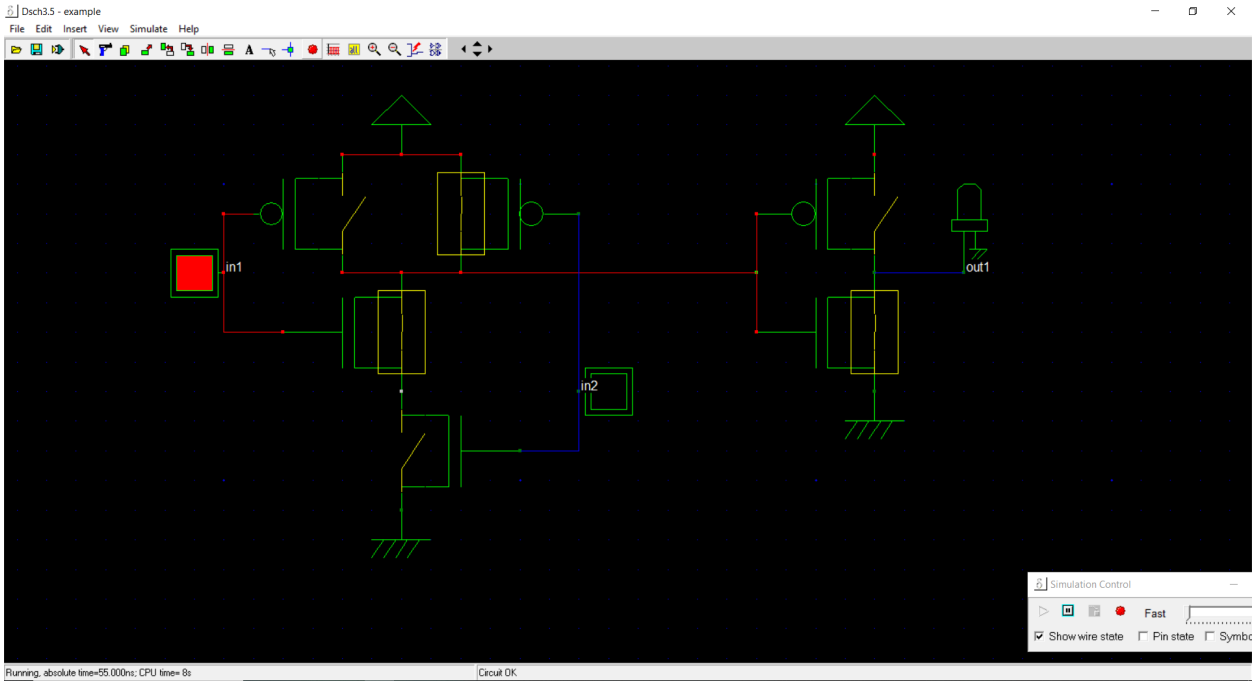
Sursa: https://www.researchgate.net/figure/The-switching-speed-of-AND-OR-XOR-and-HA-function-in-CMOS-technology_tbl1_272015389

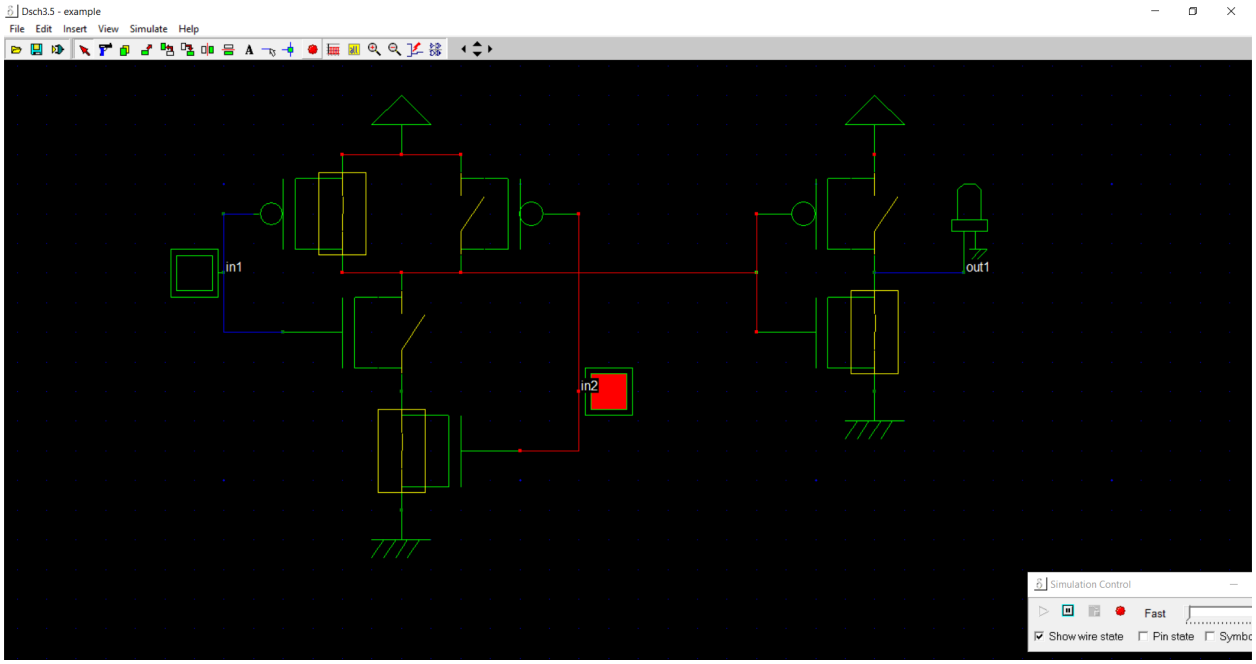
c. Implementare DSCH

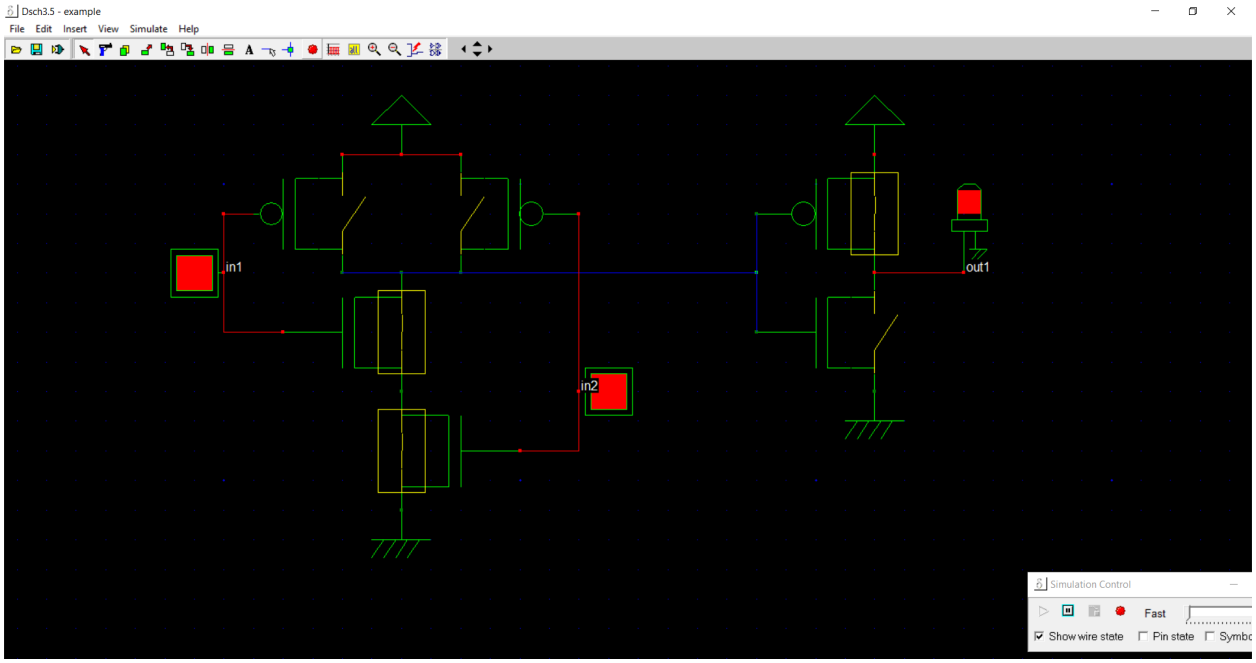


d. Simulare DSCH





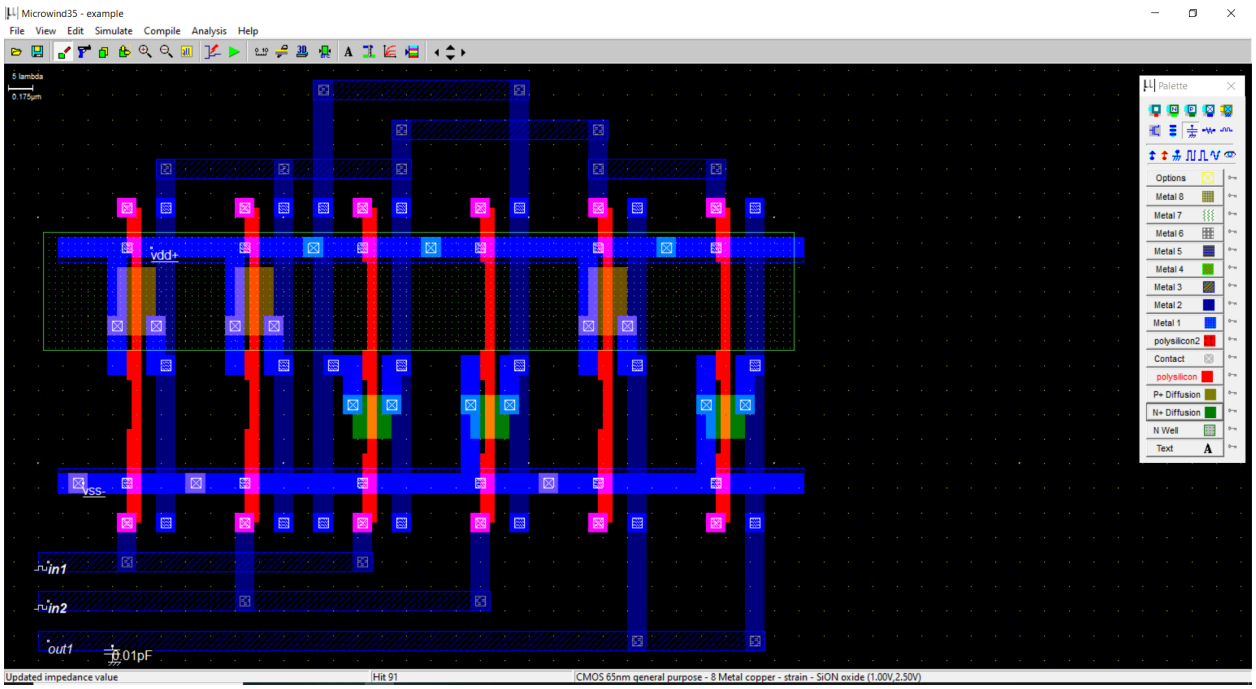

Dsch3.5 - example
File Edit Insert View Simulate Help
in1
in2
out1
Simulation Control
Fast
Show wire state
Pin state


Dsch3.5 - example
File Edit Insert View Simulate Help
in1
in2
out1
Simulation Control
Fast
Show wire state
Pin state

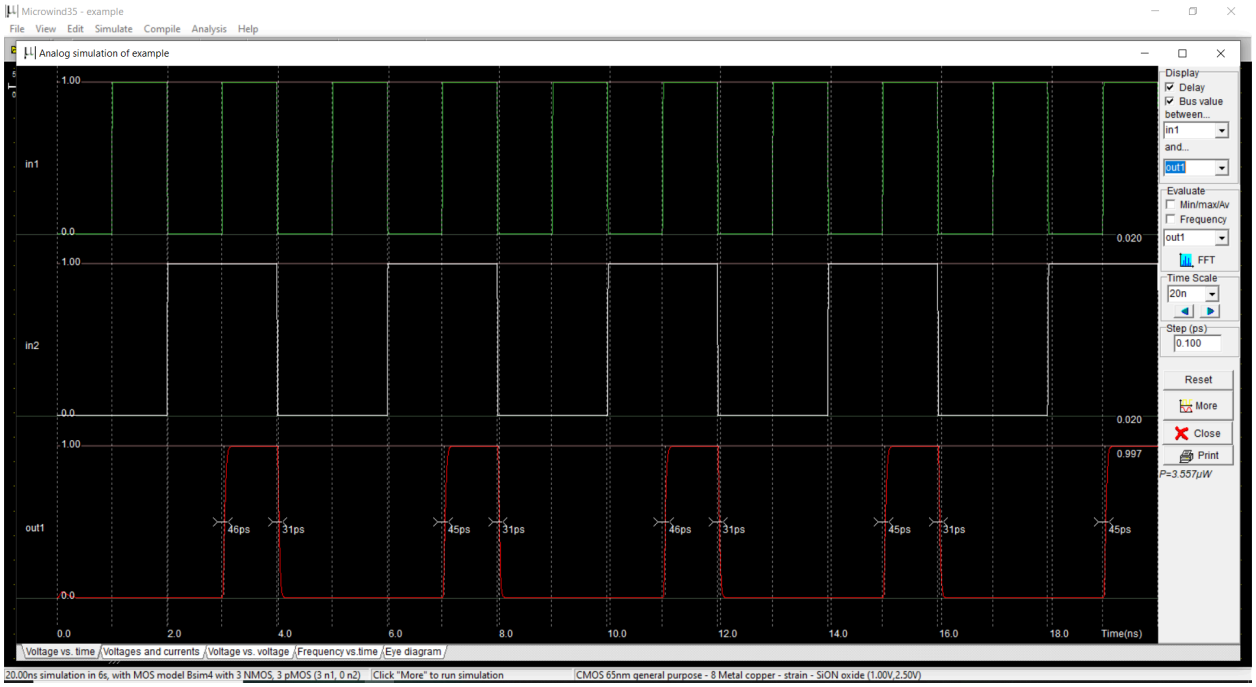
e. Cod Verilog

```
module and_dsch( in1,in2,out1);
 input in1,in2;
 output out1;
 wire w3,w5,;
 pmos #(3) pmos_1(w3,vdd,in1); // 0.5u 0.07u
 pmos #(3) pmos_2(w3,vdd,in2); // 0.5u 0.07u
 nmos #(3) nmos_3(w3,w5,in1); // 0.3u 0.07u
 nmos #(1) nmos_4(w5,vss,in2); // 0.3u 0.07u
 pmos #(2) pmos_5(out1,vdd,w3); // 0.5u 0.07u
 nmos #(2) nmos_6(out1,vss,w3); // 0.3u 0.07u
endmodule

// Simulation parameters in Verilog Format
always
#200 in1=~in1;
#400 in2=~in2;

// Simulation parameters
// in1 CLK 1 1
// in2 CLK 2 2
```

f. Implementare Microwind

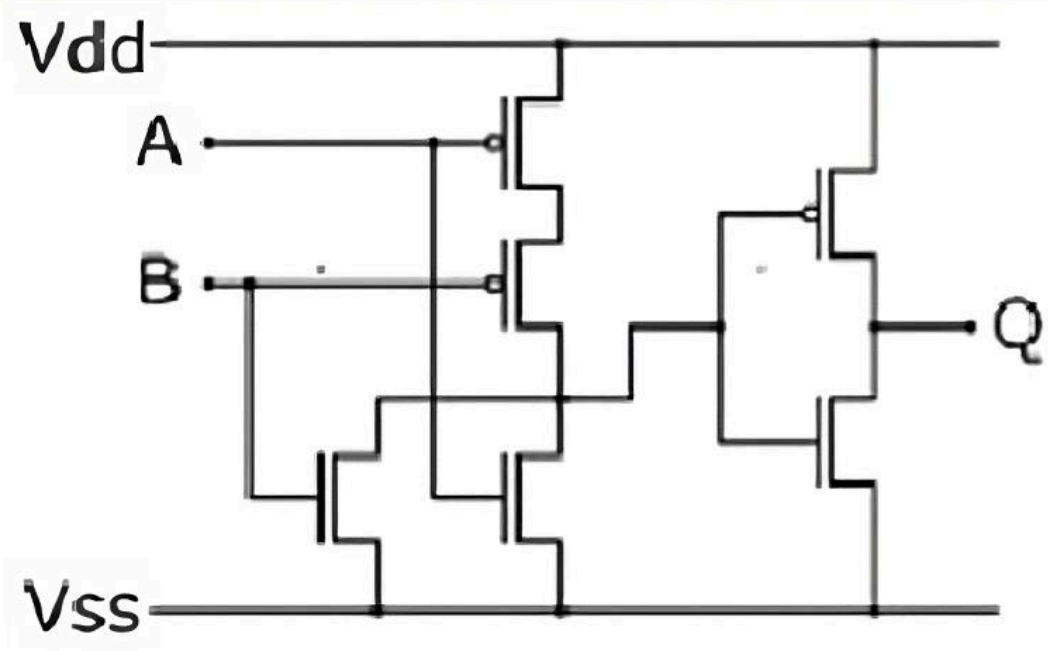


g. Simulare Microwind



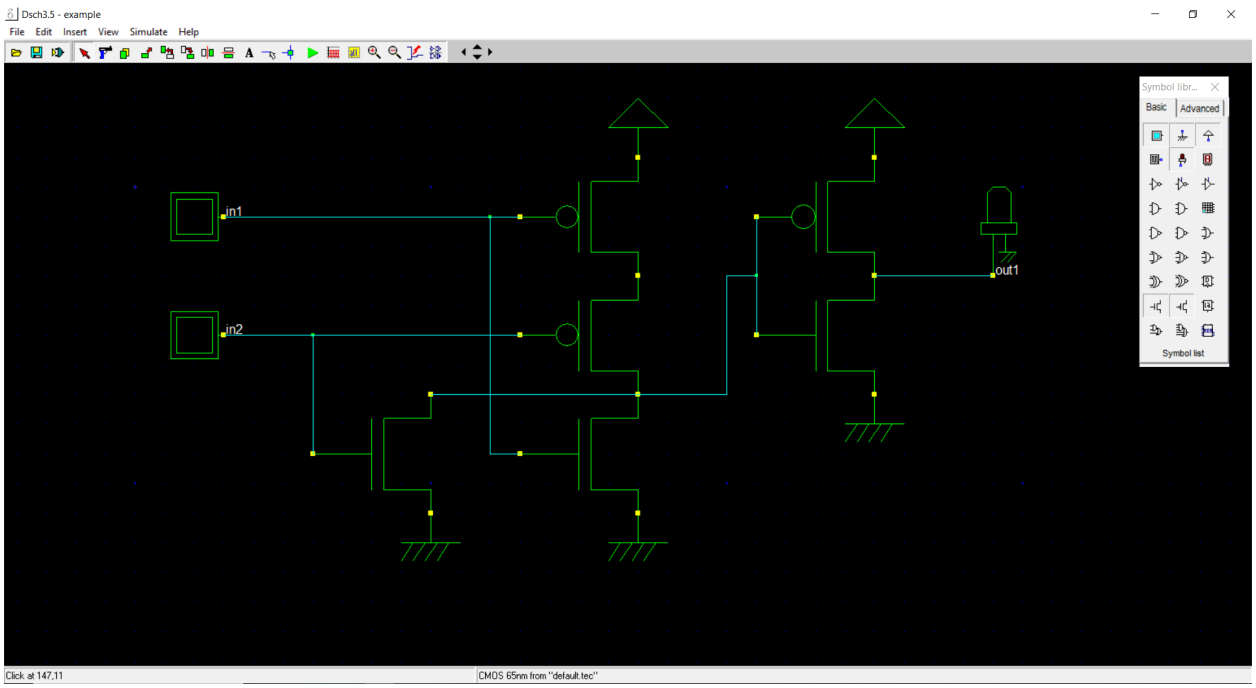
3. OR

a. Schema circuitului

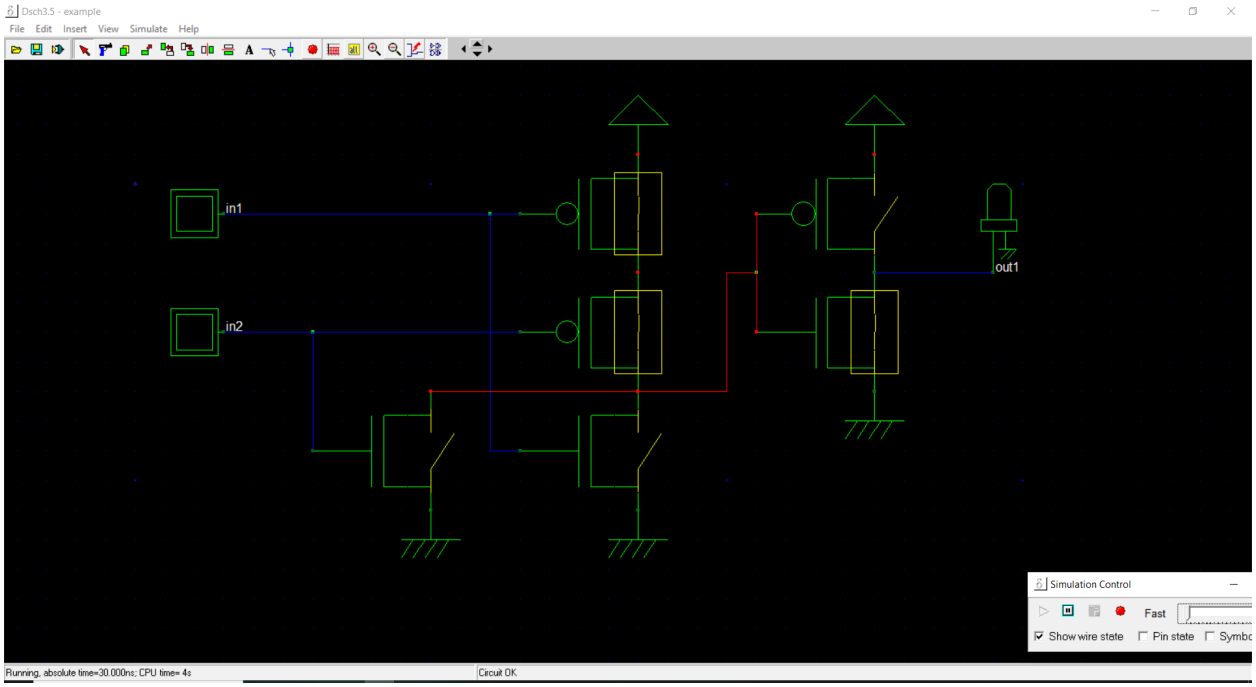


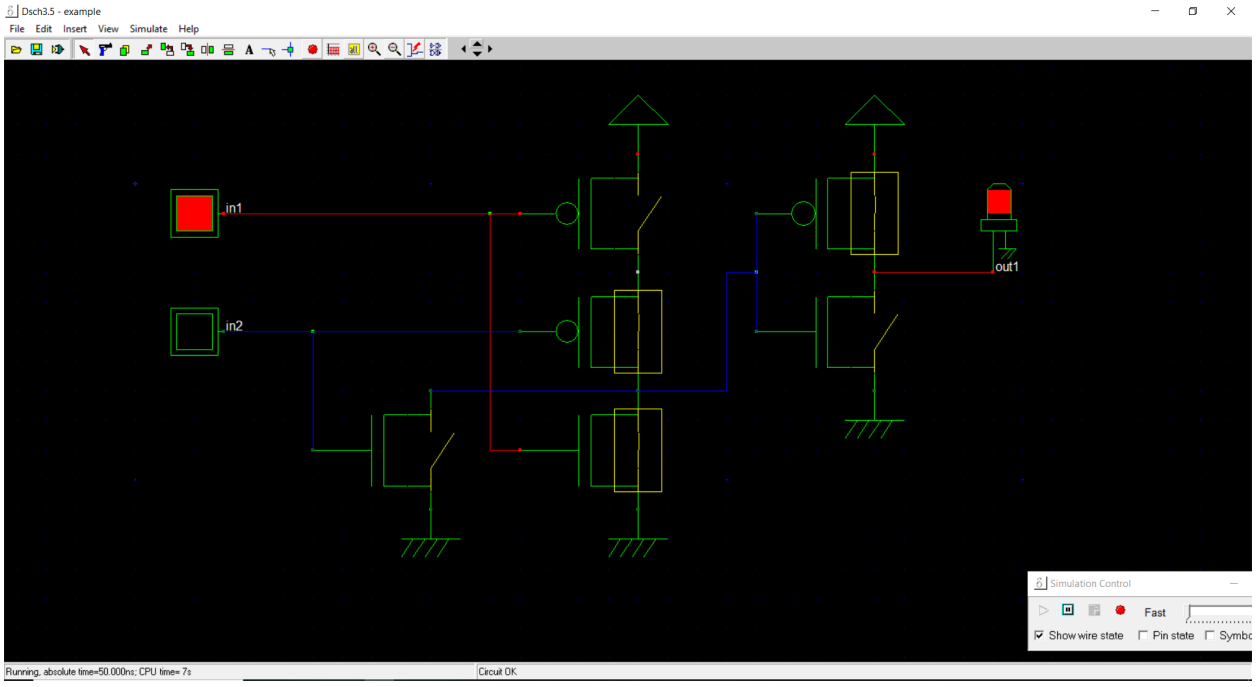
Sursa: https://www.researchgate.net/figure/The-switching-speed-of-AND-OR-XOR-and-HA-function-in-CMOS-technology_tbl1_272015389

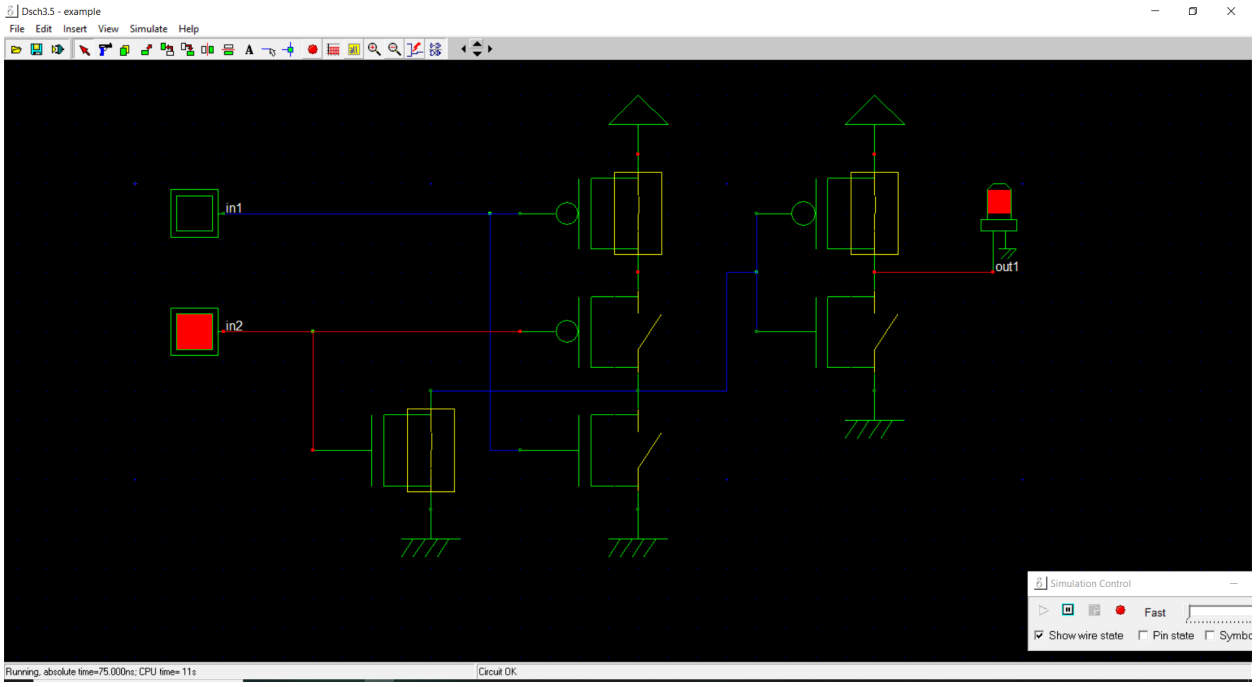
b. Implementare DSCH

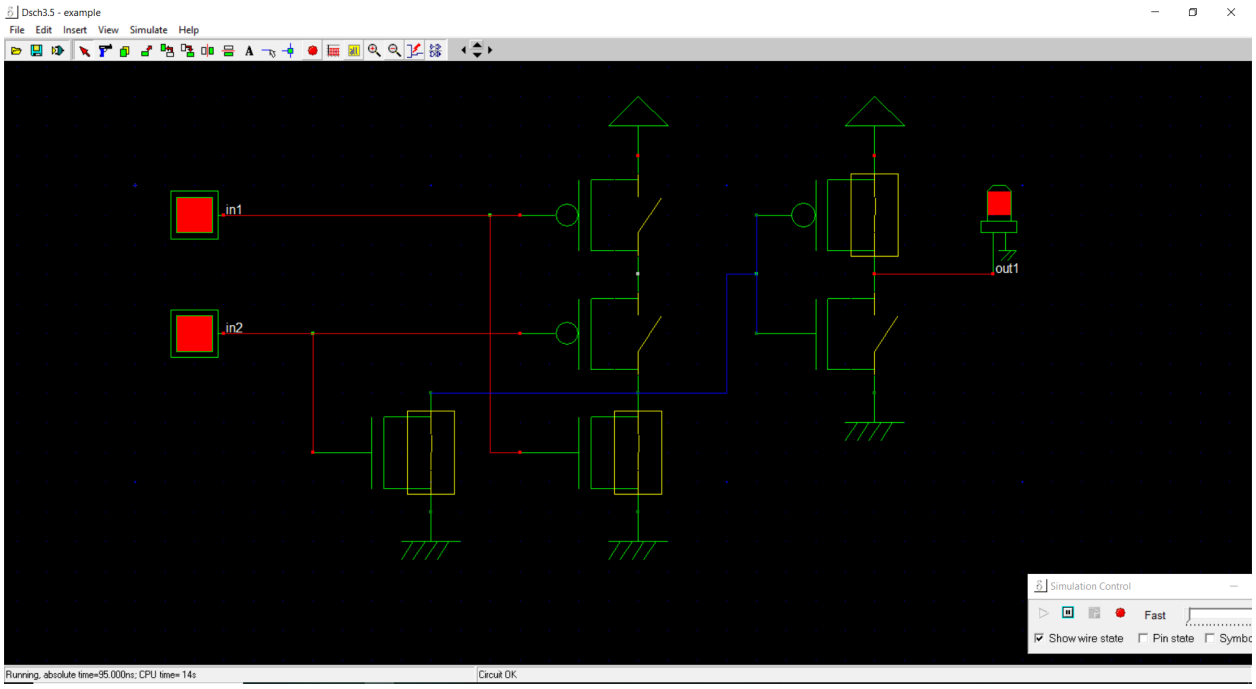


c. Simulare DSCH





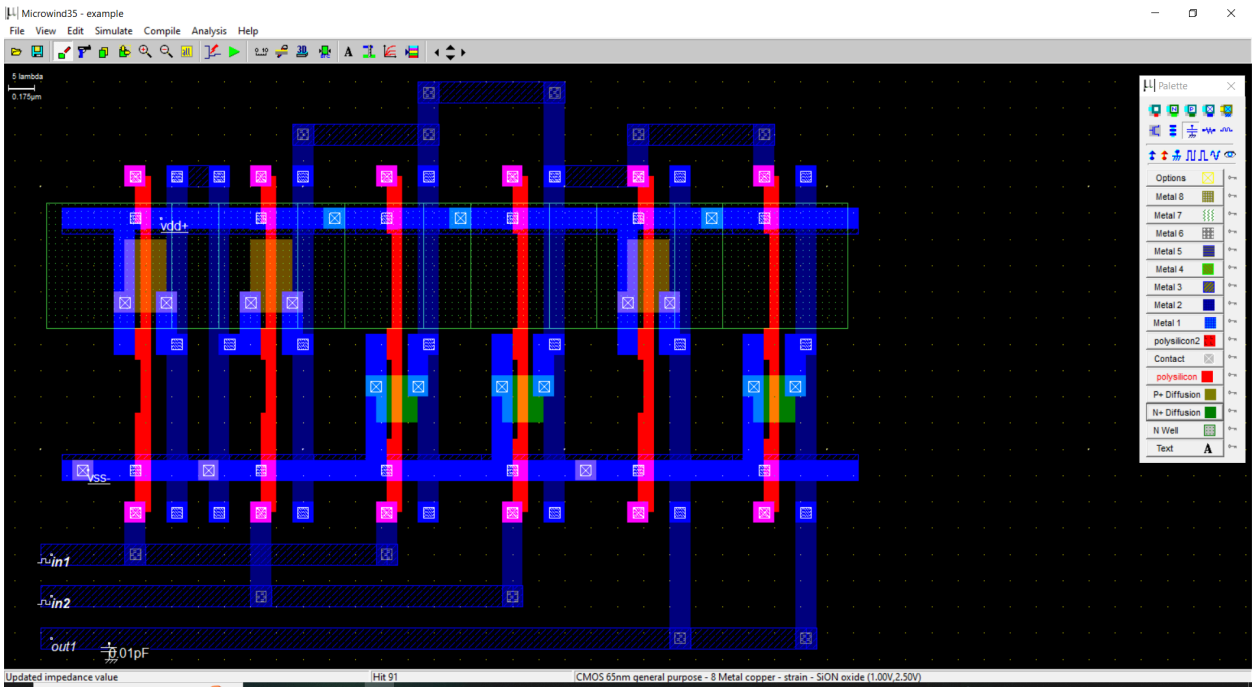

Dsch3.5 - example
File Edit Insert View Simulate Help
in1
in2
out1
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=75.000ns; CPU time= 11s
Circuit OK


Dsch3.5 - example
File Edit Insert View Simulate Help
in1
in2
out1
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=95.000ns; CPU time= 14s
Circuit OK

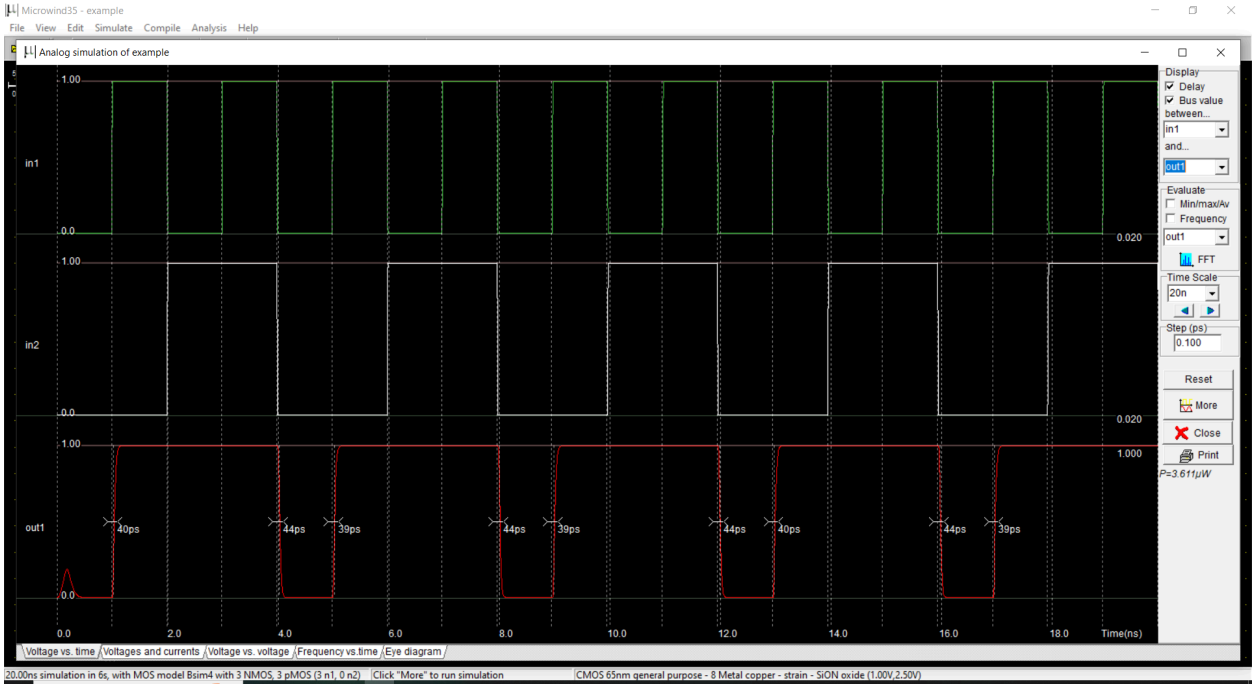
d. Cod Verilog

```
module example( in1,in2,out1);
 input in1,in2;
 output out1;
 wire w3,w5,;
 pmos #(1) pmos_1(w3,vdd,in1); // 0.5u 0.07u
 pmos #(3) pmos_2(w5,w3,in2); // 0.5u 0.07u
 nmos #(3) nmos_3(w5,vss,in1); // 0.3u 0.07u
 nmos #(3) nmos_4(w5,vss,in2); // 0.3u 0.07u
 pmos #(2) pmos_5(out1,vdd,w5); // 0.5u 0.07u
 nmos #(2) nmos_6(out1,vss,w5); // 0.3u 0.07u
endmodule

// Simulation parameters in Verilog Format
always
#200 in1=~in1;
#400 in2=~in2;

// Simulation parameters
// in1 CLK 1 1
// in2 CLK 2 2
```

e. Implementare Microwind

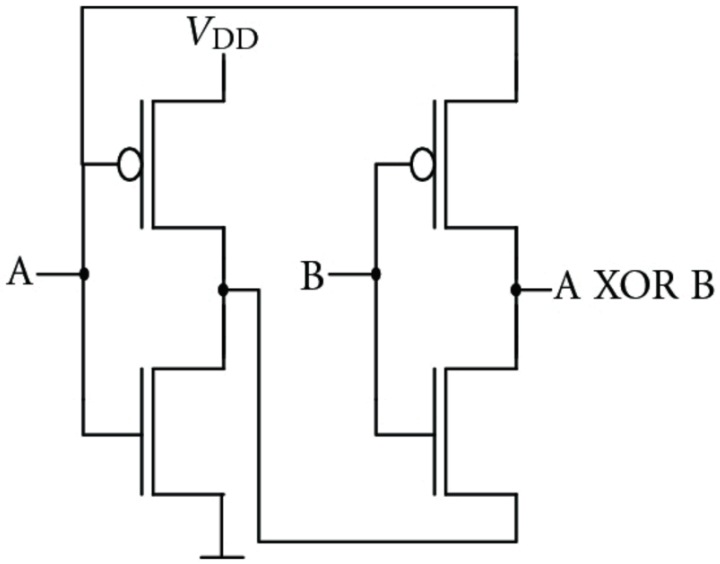


f. Simulare Microwind



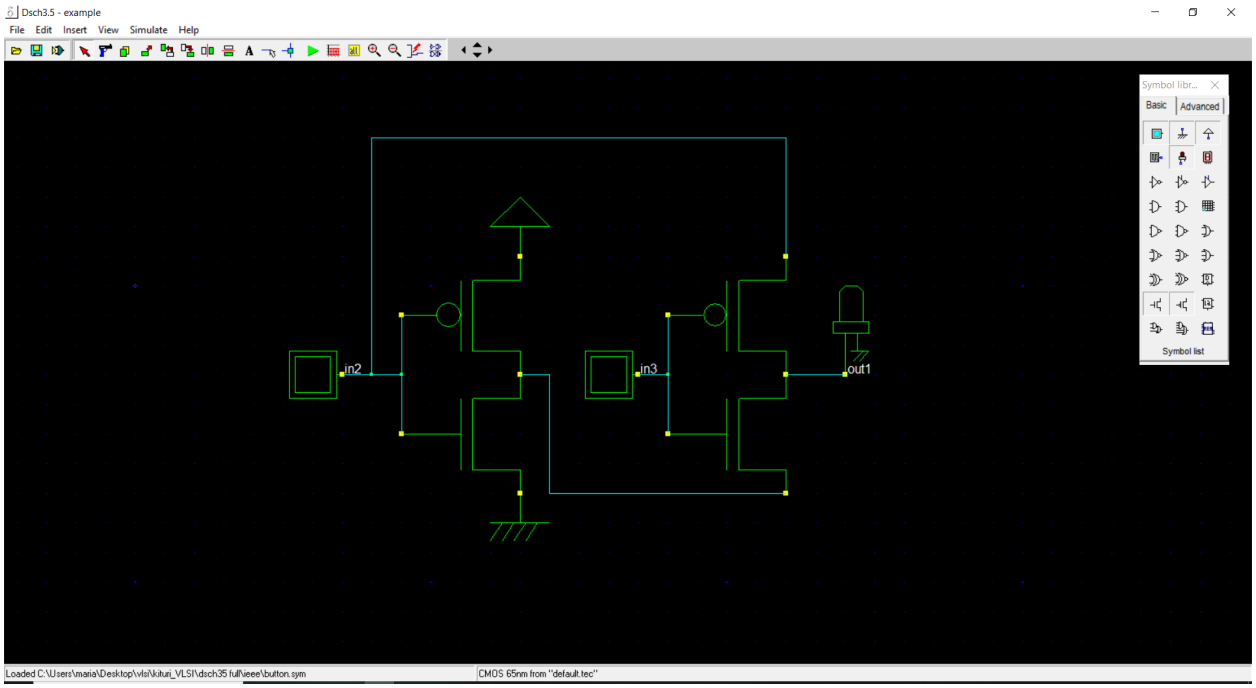
4. XOR

a. Schema circuitului

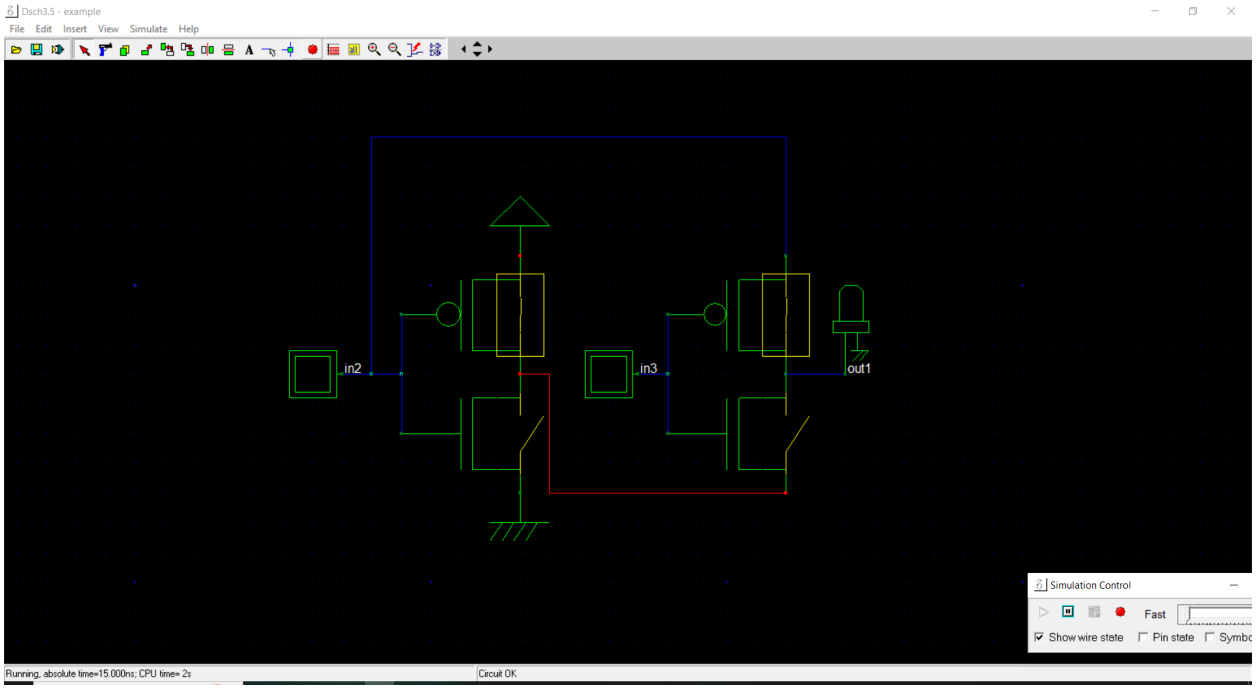


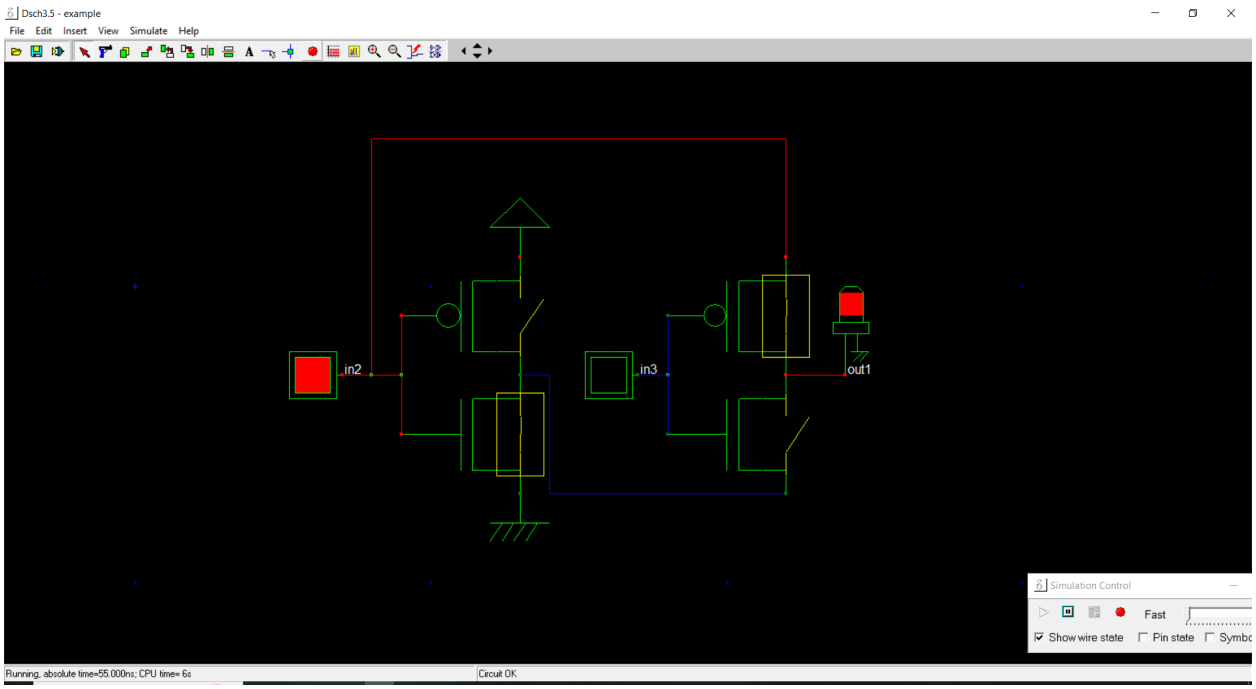
Sursa: https://www.researchgate.net/figure/Basic-designs-of-XOR-XNOR-gate-found-in-literature_fig10_228534271

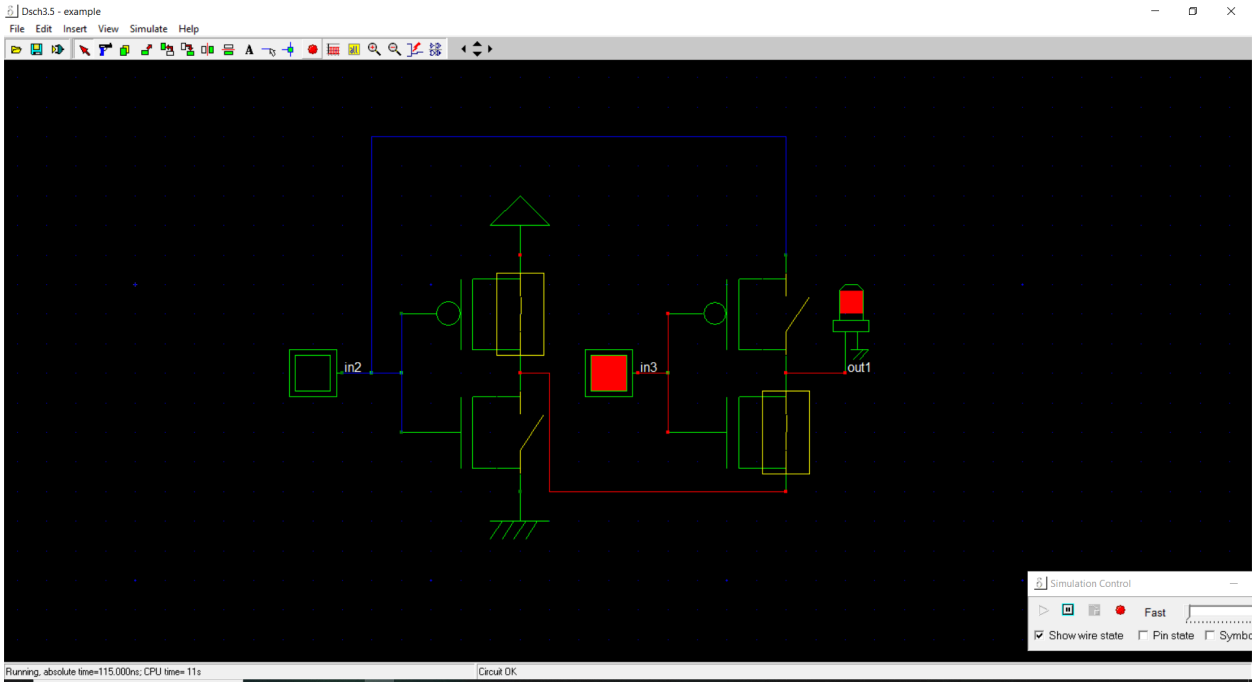
b. Implementare DSCH

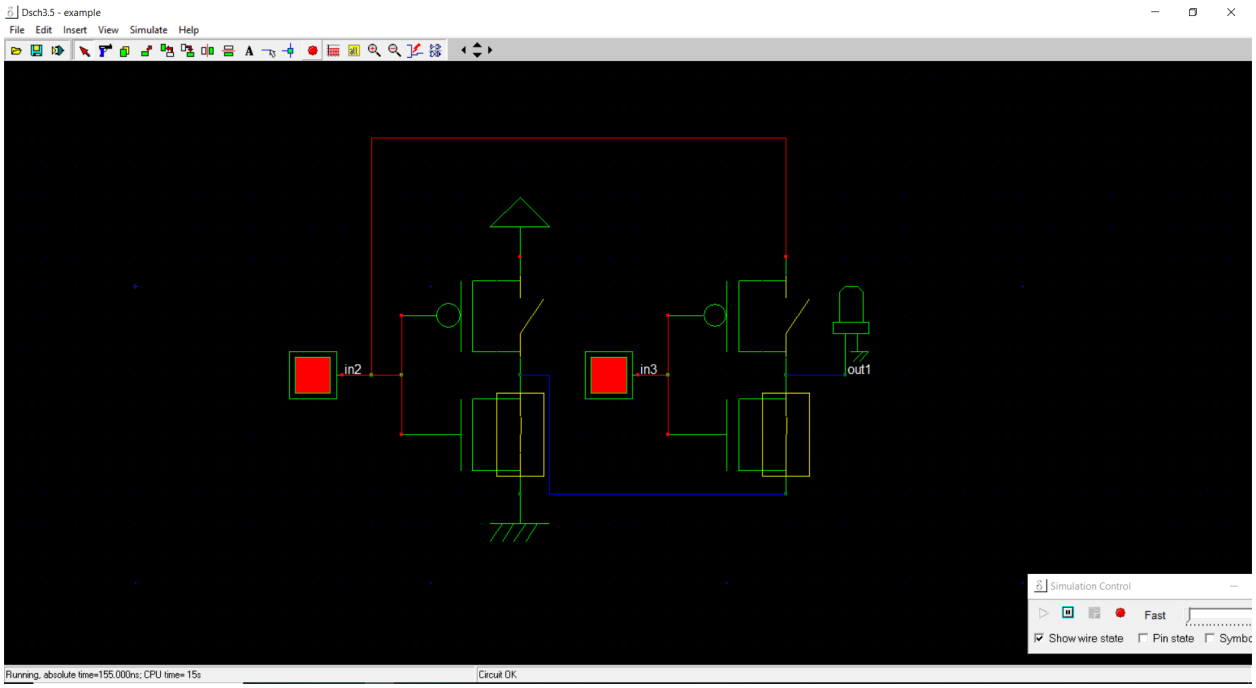


c. Simulare DSCH





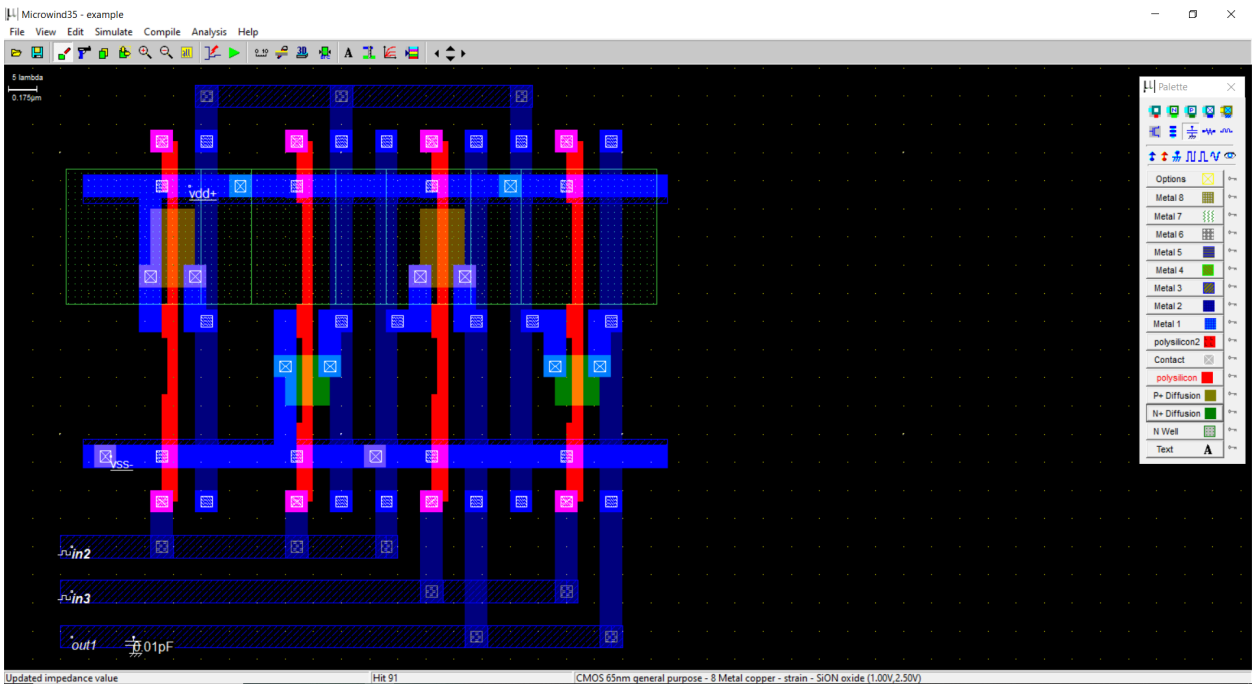

Dsch3.5 - example
File Edit Insert View Simulate Help
in2
in3
out1
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=115.000ns; CPU time= 11s
Circuit OK


Dsch3.5 - example
File Edit Insert View Simulate Help
in2
in3
out1
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=155.000ns; CPU time= 15s
Circuit OK

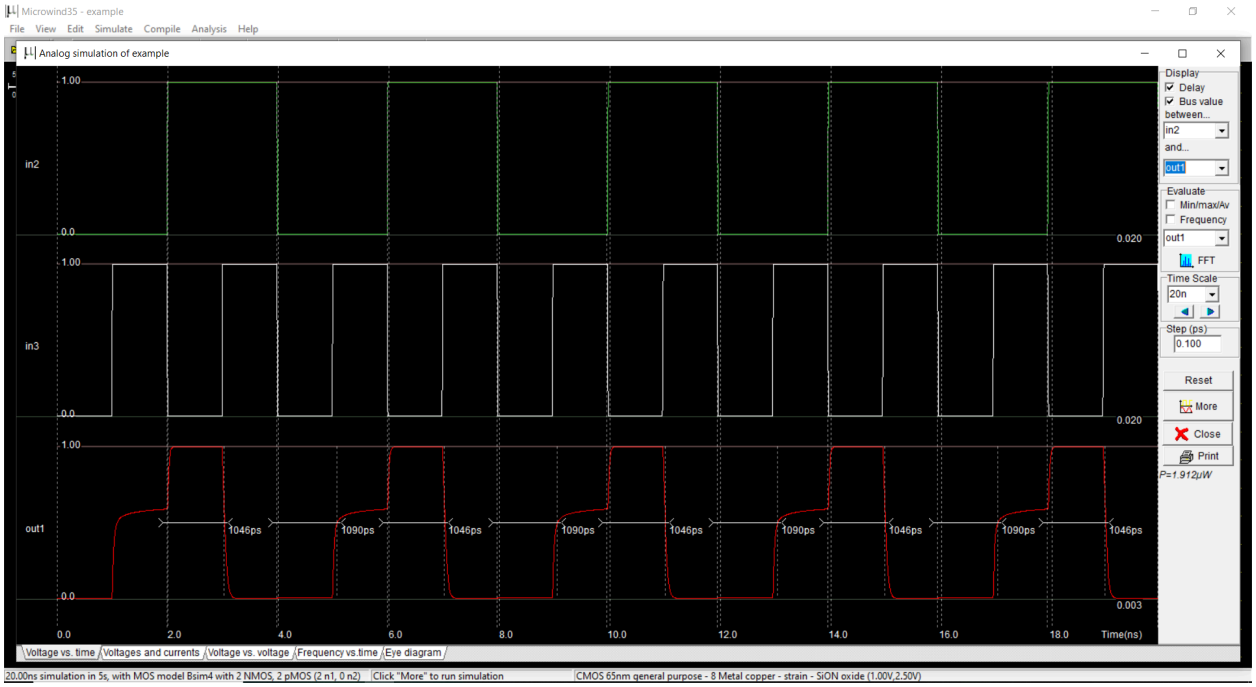
d. Cod Verilog

```
module example( in3,in2,out1);
 input in3,in2;
 output out1;
 wire w3,;
 pmos #(2) pmos_1(w3,vdd,in2); // 0.5u 0.07u
 nmos #(2) nmos_2(w3,vss,in2); // 0.3u 0.07u
 pmos #(2) pmos_3(out1,in2,in3); // 0.5u 0.07u
 nmos #(2) nmos_4(out1,w3,in3); // 0.3u 0.07u
endmodule

// Simulation parameters in Verilog Format
always
#200 in3=~in3;
#400 in2=~in2;

// Simulation parameters
// in3 CLK 1 1
// in2 CLK 2 2
```

e. Implementare Microwind

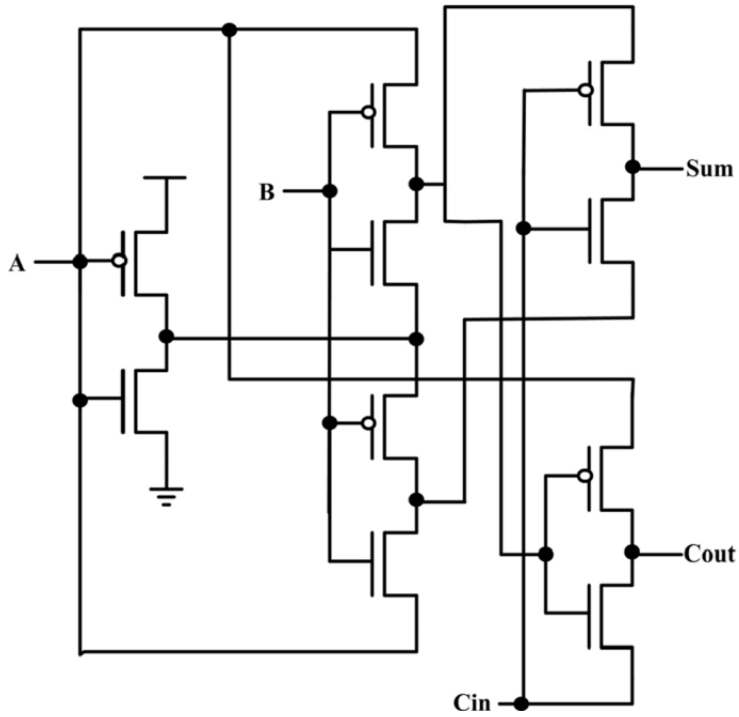


f. Simulare Microwind



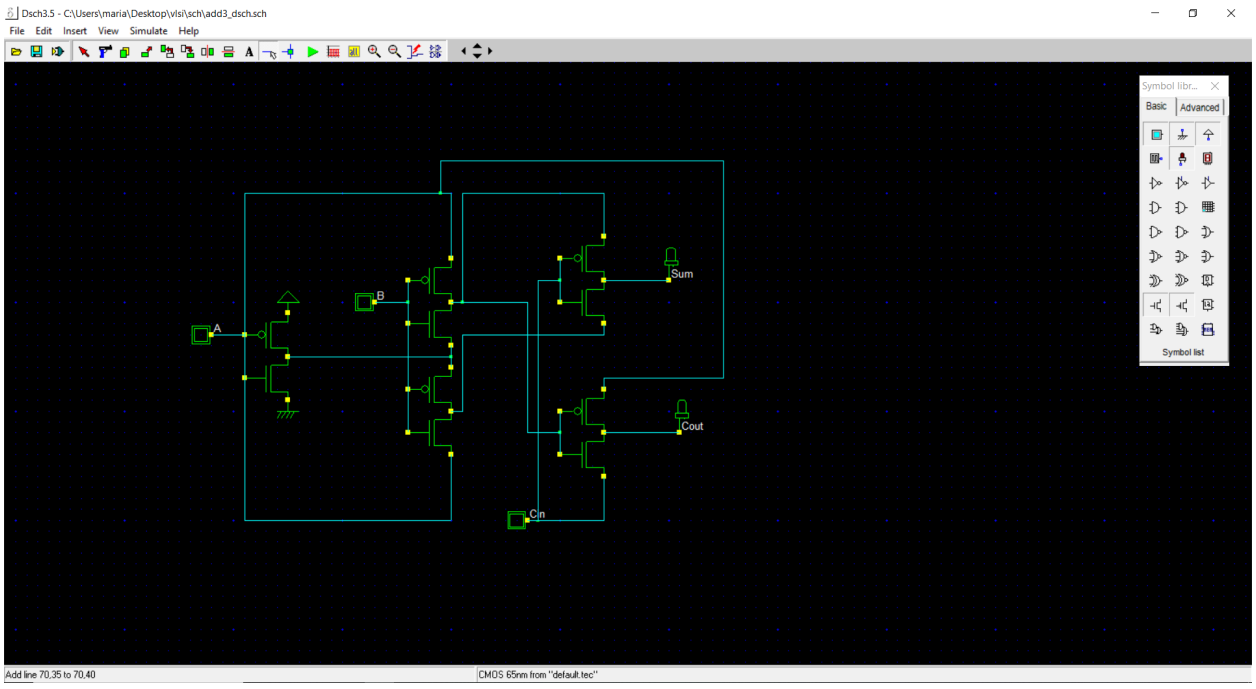
5. Sumator

a. Schema circuitului

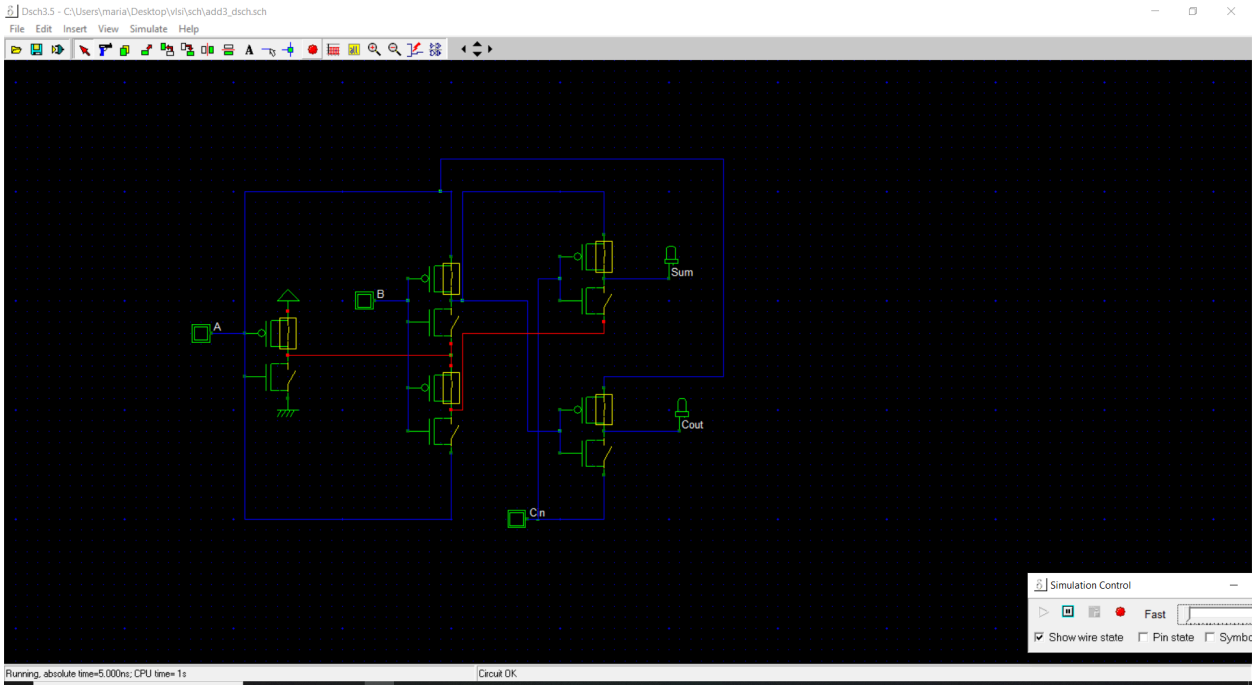


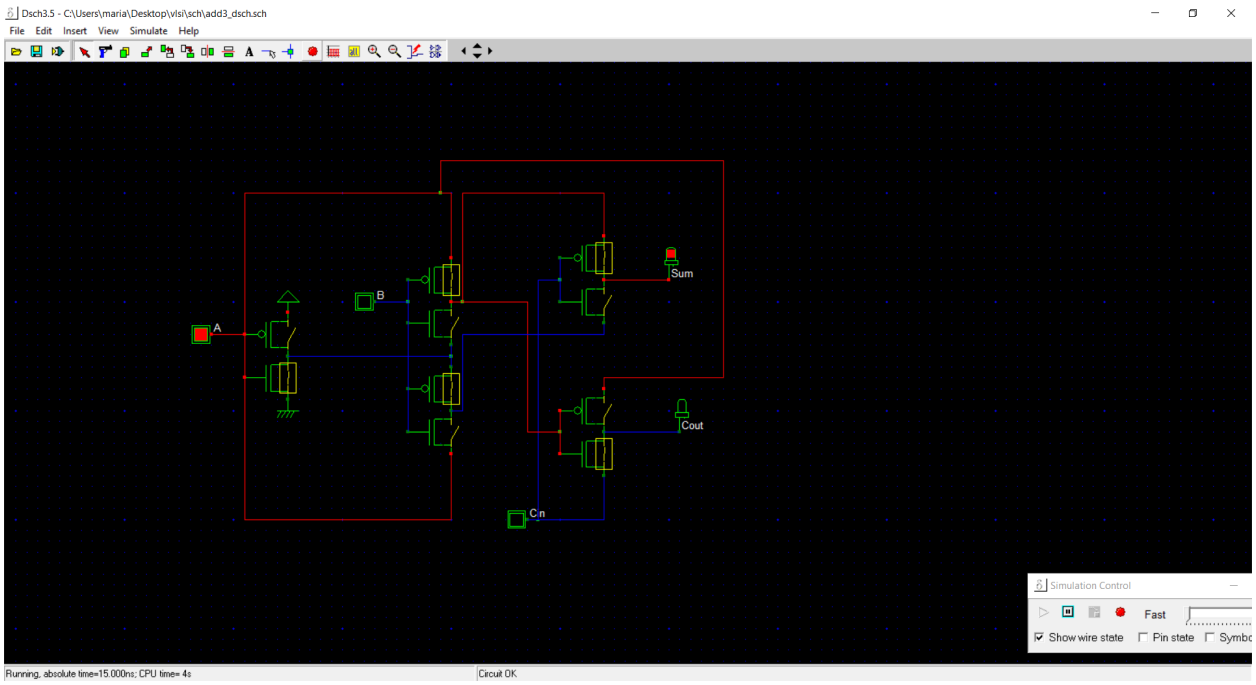
Sursa: https://link.springer.com/article/10.1007/s00034-020-01550-3

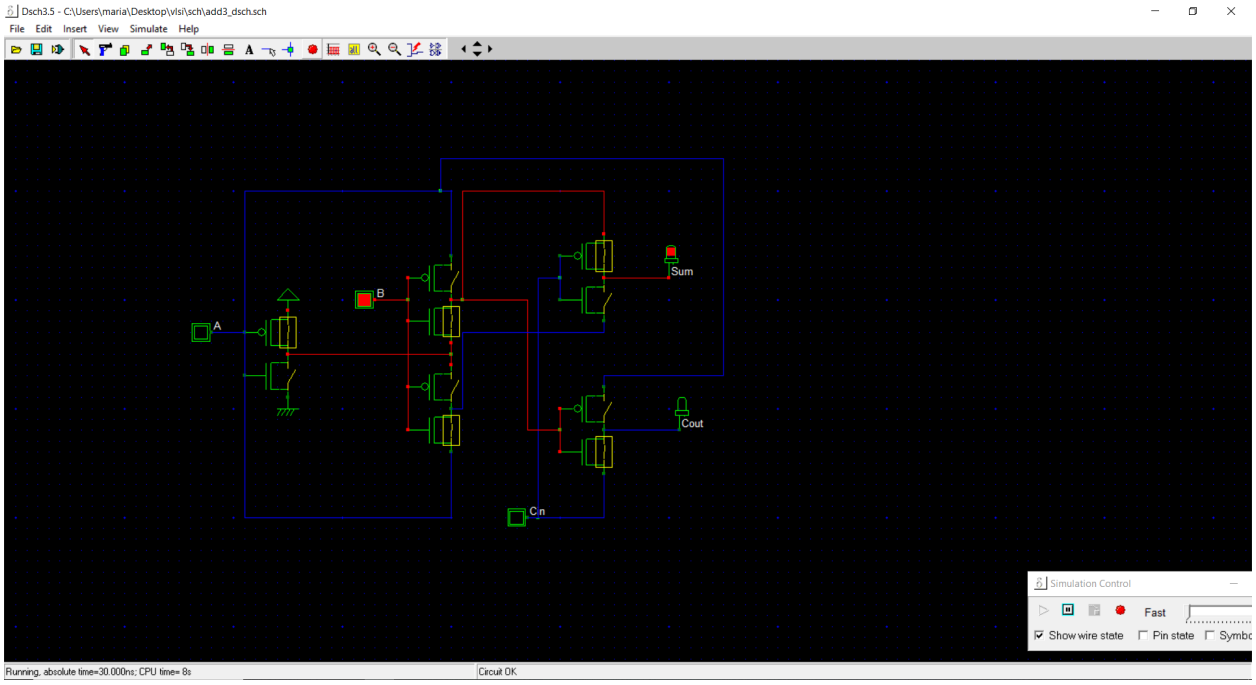
b. Implementare DSCH

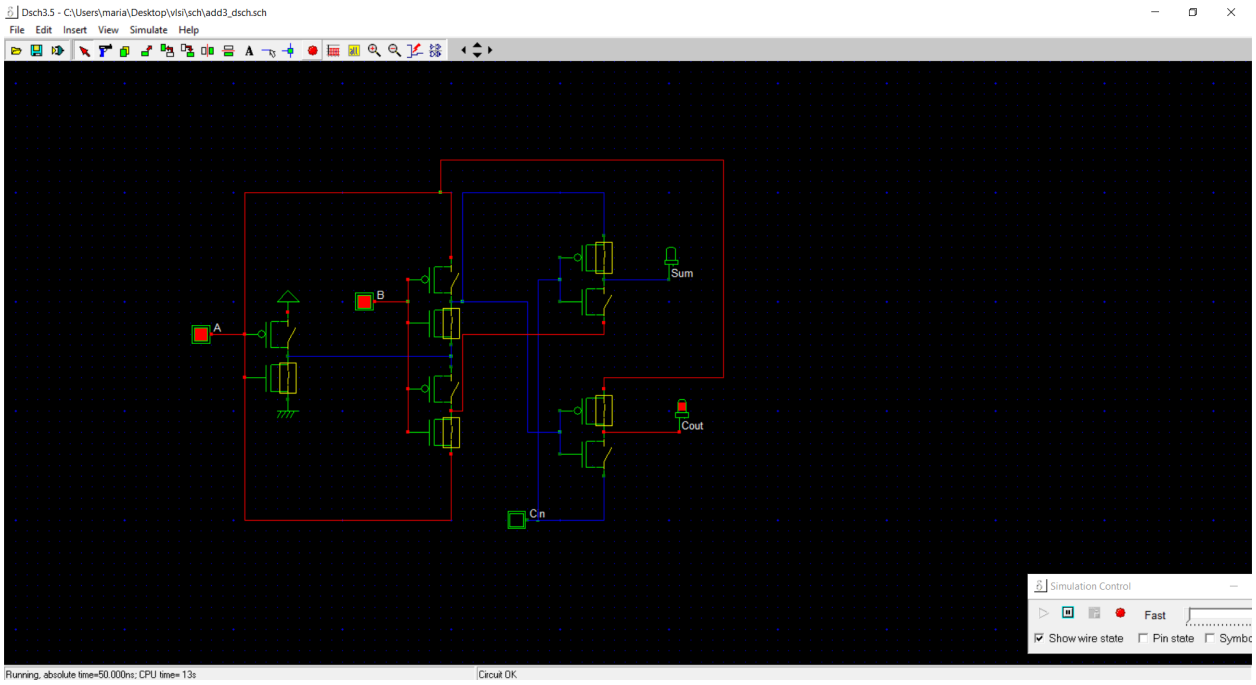


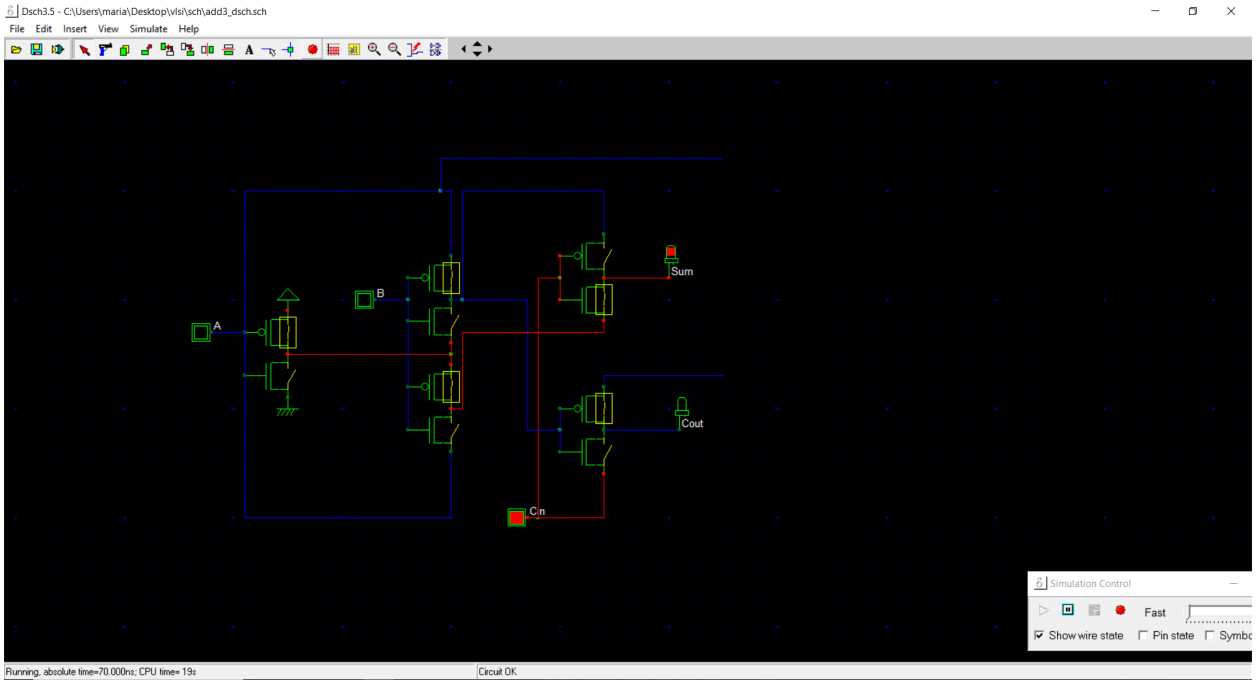
c. Simulare DSCH

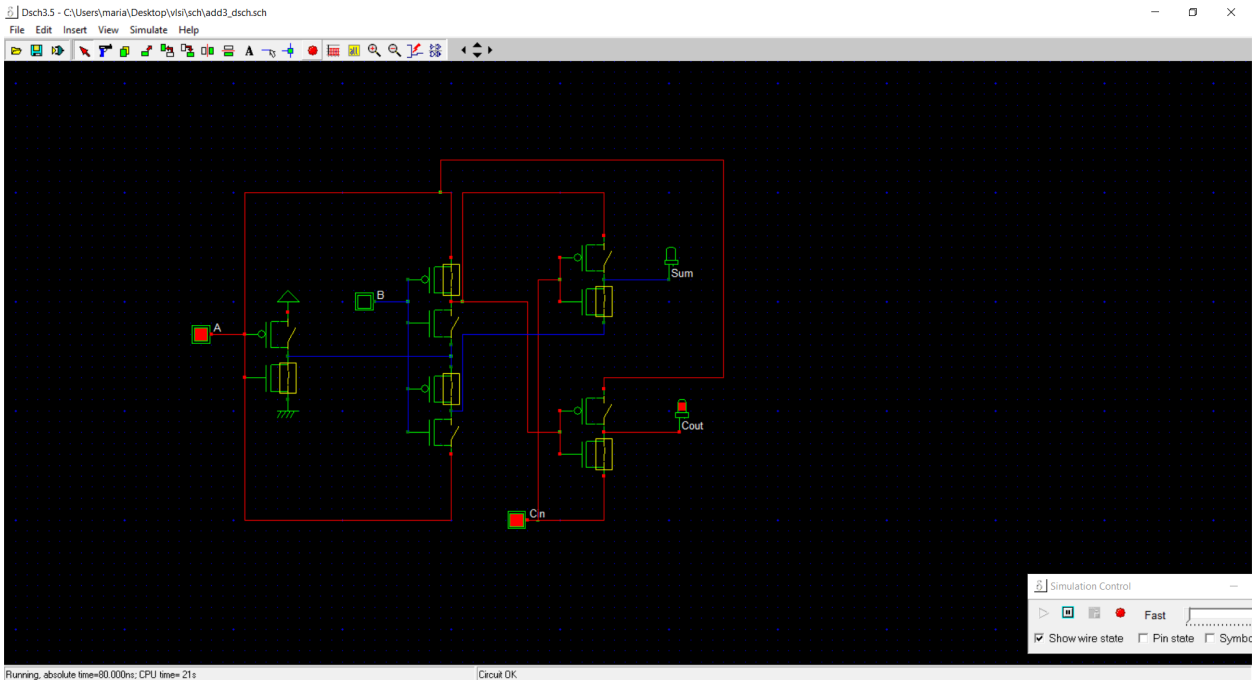


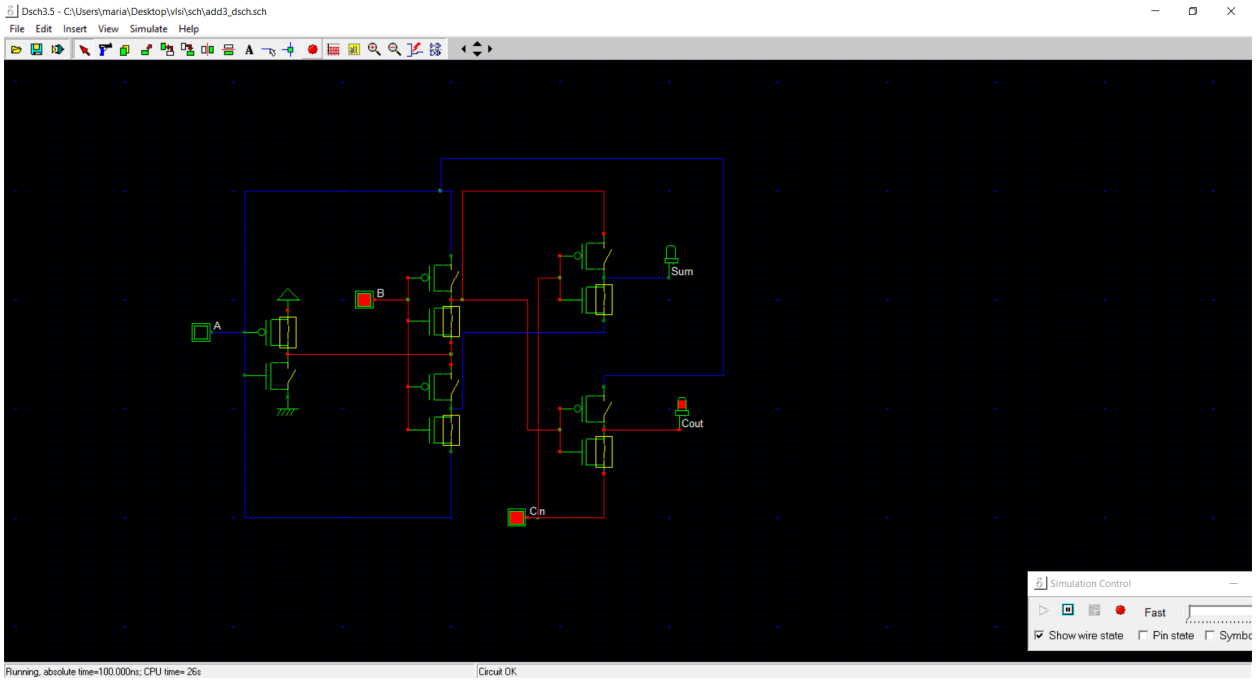


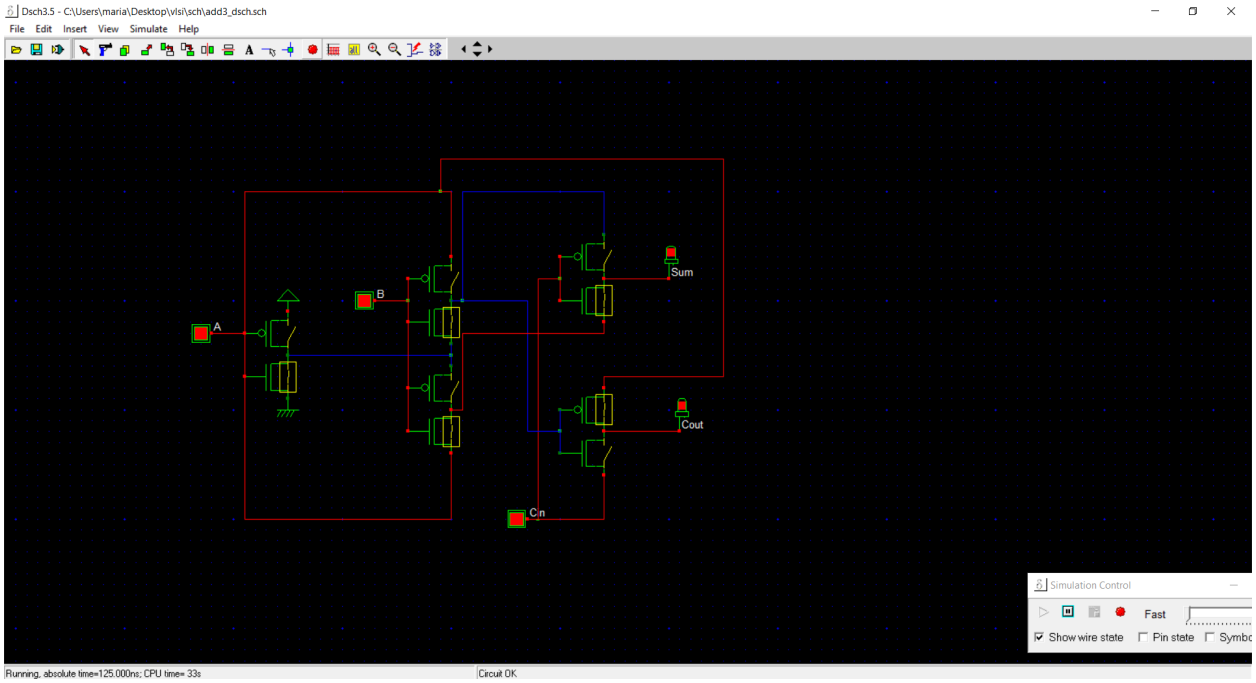

Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\add3_dsch.sch
File Edit Insert View Simulate Help
A
B
Cin
Sum
Cout
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=30.000ns; CPU time= 8s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\add3_dsch.sch
File Edit Insert View Simulate Help
A
B
Cin
Sum
Cout
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=50.000ns; CPU time= 13s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\add3_dsch.sch
File Edit Insert View Simulate Help
A
B
Sum
Cout
Cin
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=70.000ns; CPU time= 19s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\add3_dsch.sch
File Edit Insert View Simulate Help
A
B
Sum
Cout
Cin
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=80.000ns; CPU time= 21s
Circuit OK

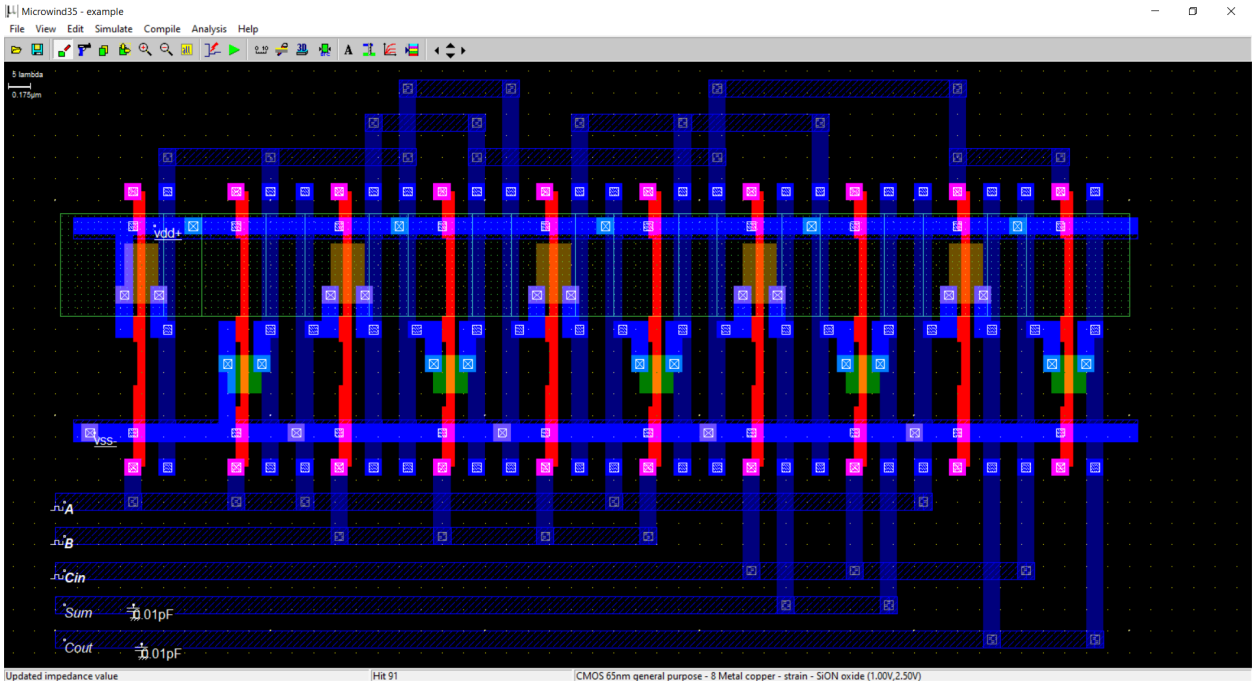

Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\add3_dsch.sch
File Edit Insert View Simulate Help
A
B
Sum
Cout
Cin
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=100.000ns; CPU time= 26s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\add3_dsch.sch
File Edit Insert View Simulate Help
A
B
Sum
Cout
Cin
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=125.000ns; CPU time= 33s
Circuit OK

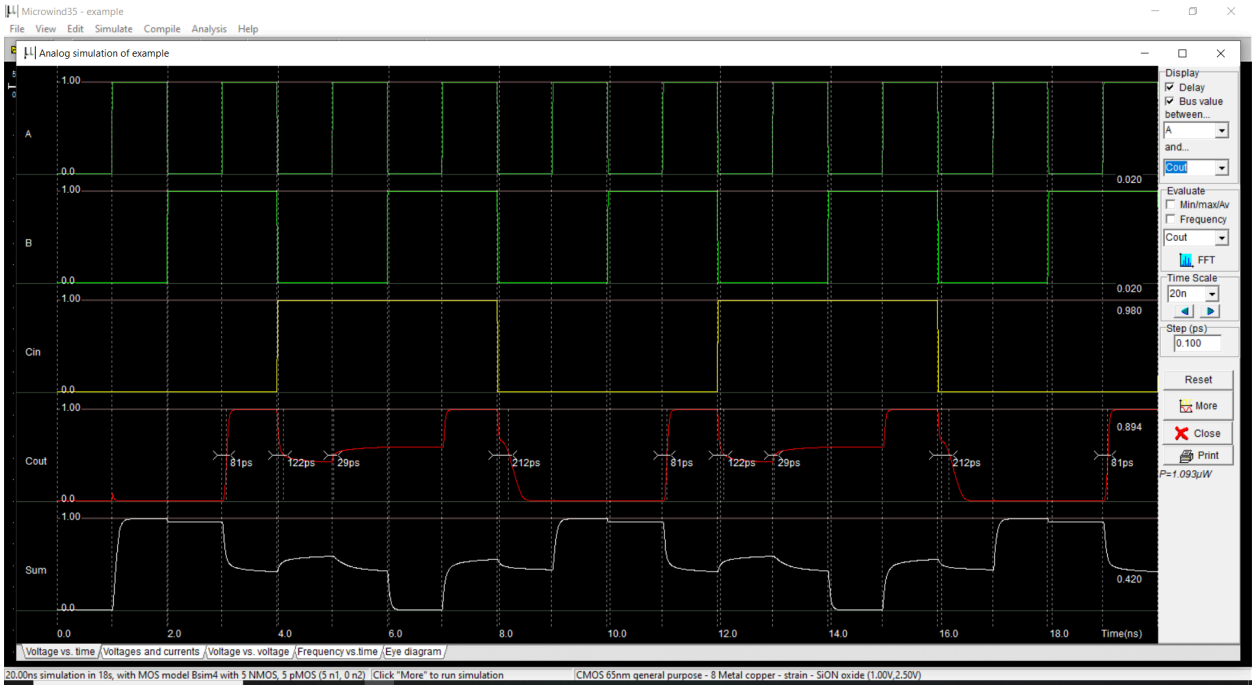
d. Cod Verilog

```
module add_dsch( A,B,Cin,Sum,Cout);
 input A,B,Cin;
 output Sum,Cout;
 wire w3,w5,w6,;
 pmos #(2) pmos_1(w3,vdd,A); // 0.5u 0.07u
 nmos #(2) nmos_2(w3,vss,A); // 0.3u 0.07u
 pmos #(3) pmos_3(w5,A,B); // 0.5u 0.07u
 nmos #(3) nmos_4(w5,w3,B); // 0.3u 0.07u
 pmos #(2) pmos_5(w6,w3,B); // 0.5u 0.07u
 nmos #(2) nmos_6(w6,A,B); // 0.3u 0.07u
 pmos #(2) pmos_7(Sum,w5,Cin); // 0.5u 0.07u
 nmos #(2) nmos_8(Sum,w6,Cin); // 0.3u 0.07u
 pmos #(2) pmos_9(Cout,A,w5); // 0.5u 0.07u
 nmos #(2) nmos_10(Cout,Cin,w5); // 0.3u 0.07u
endmodule

// Simulation parameters in Verilog Format
always
#200 A=~A;
#400 B=~B;
#800 Cin=~Cin;

// Simulation parameters
// A CLK 1 1
// B CLK 2 2
// Cin CLK 4 4
```

e. Implementare Microwind

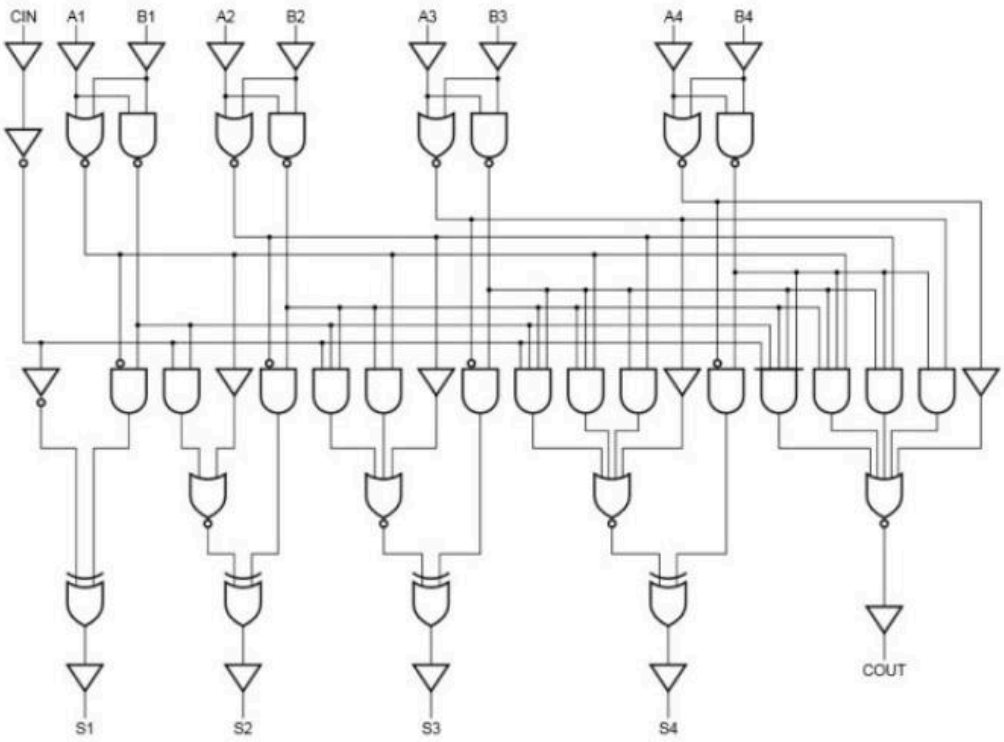


f. Simulare Microwind



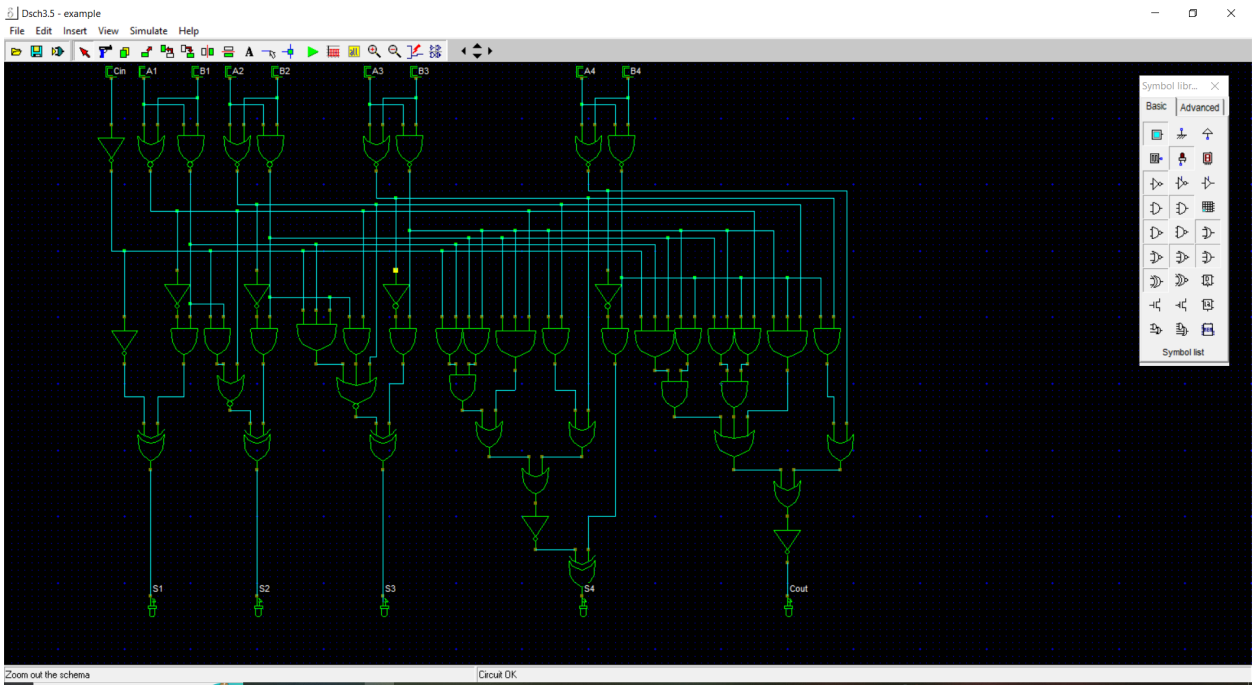
6. Sumator 4 biți Carry Look Ahead

a. Schema circuitului

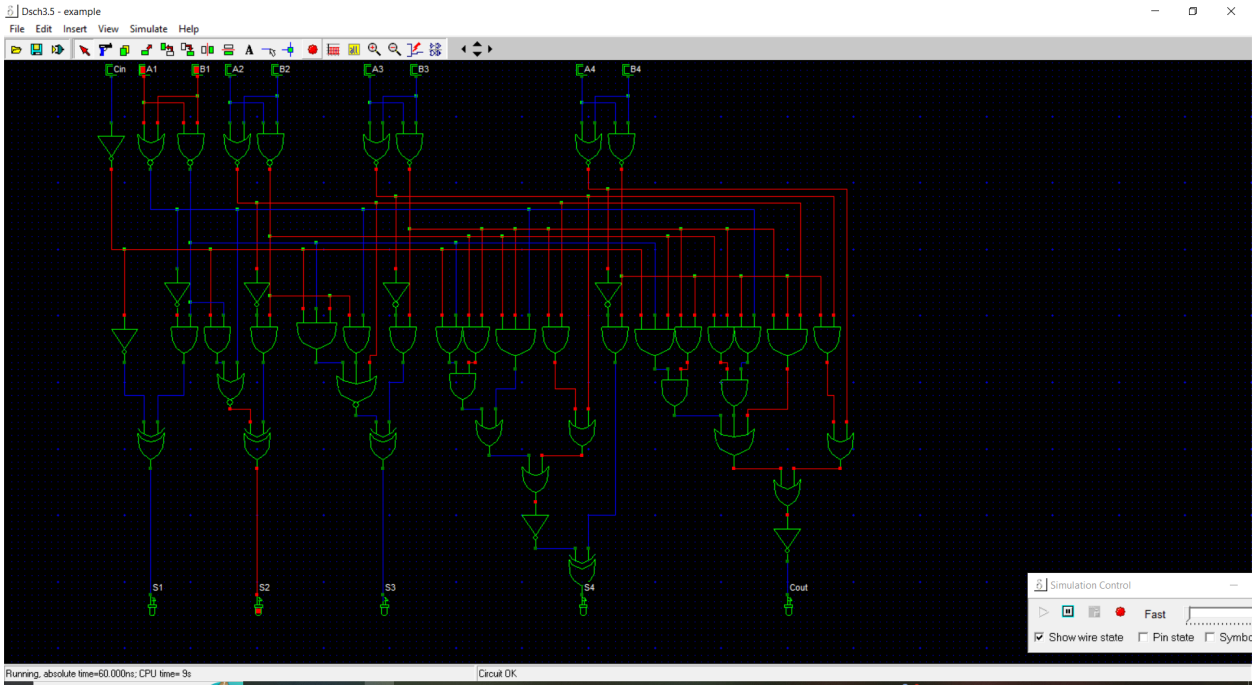


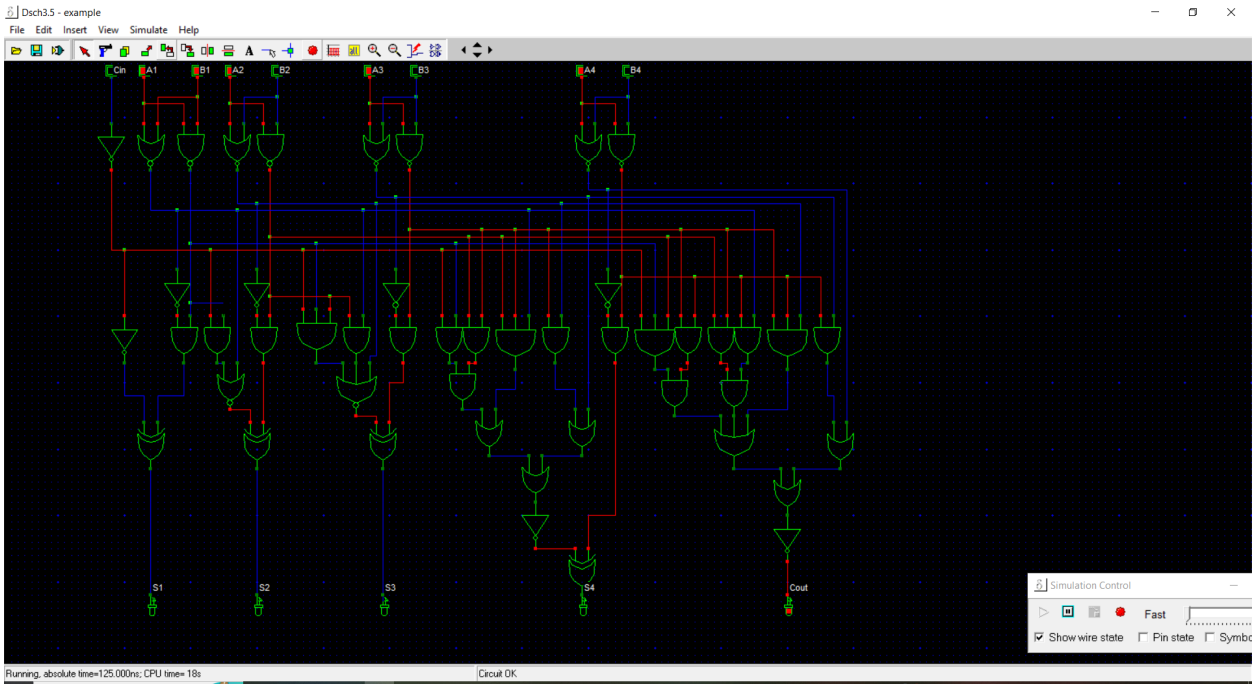
Sursa: https://www.elprocus.com/carry-look-ahead-adder/

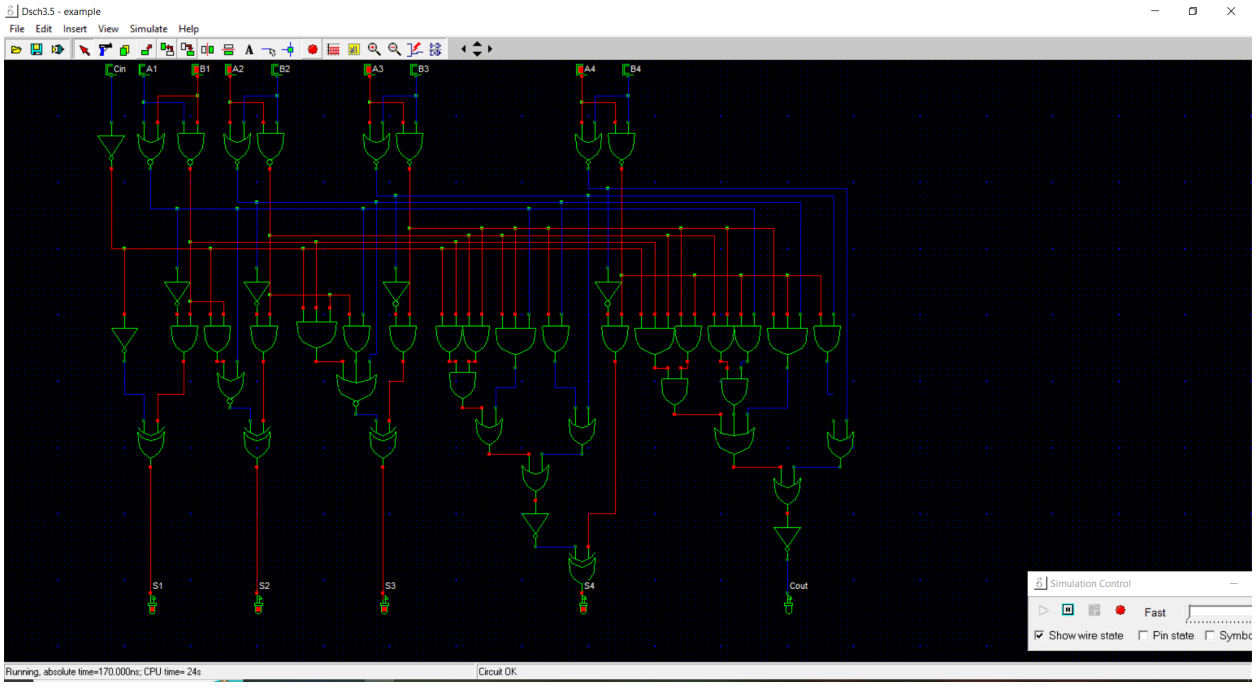
b. Implementare DSCH

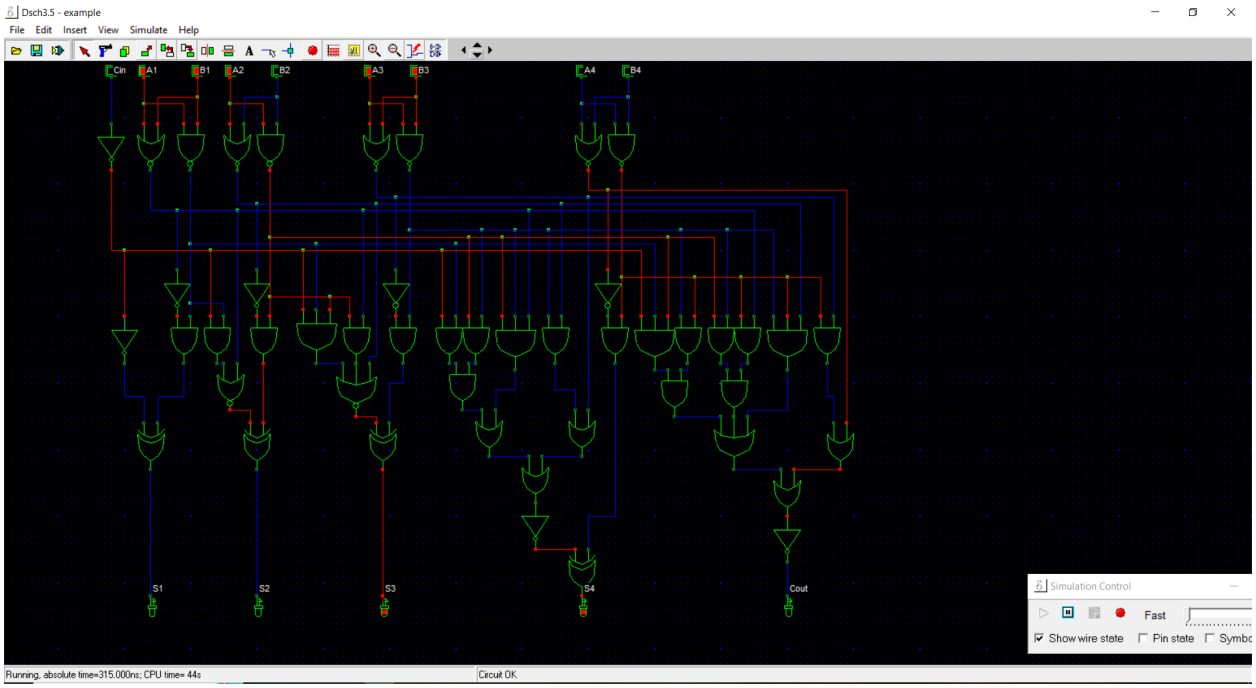


c. Simulare DSCH





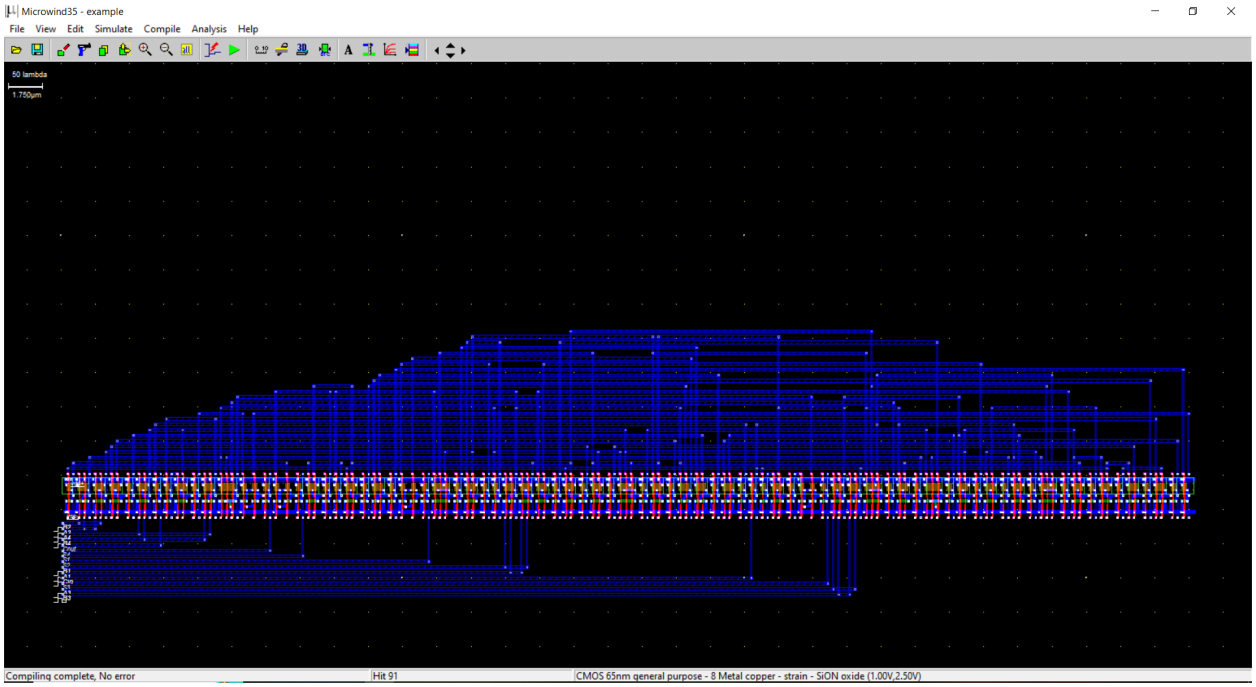

Dsch3.5 - example
File Edit Insert View Simulate Help
Cin A1 B1 A2 B2 A3 B3 A4 B4
S1 S2 S3 S4 Cout
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=170.000ns; CPU time= 24s
Circuit OK


Dsch3.5 - example
File Edit Insert View Simulate Help
Cin A1 B1 A2 B2 A3 B3 A4 B4
S1 S2 S3 S4 Cout
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=315.000ns; CPU time= 44s
Circuit OK

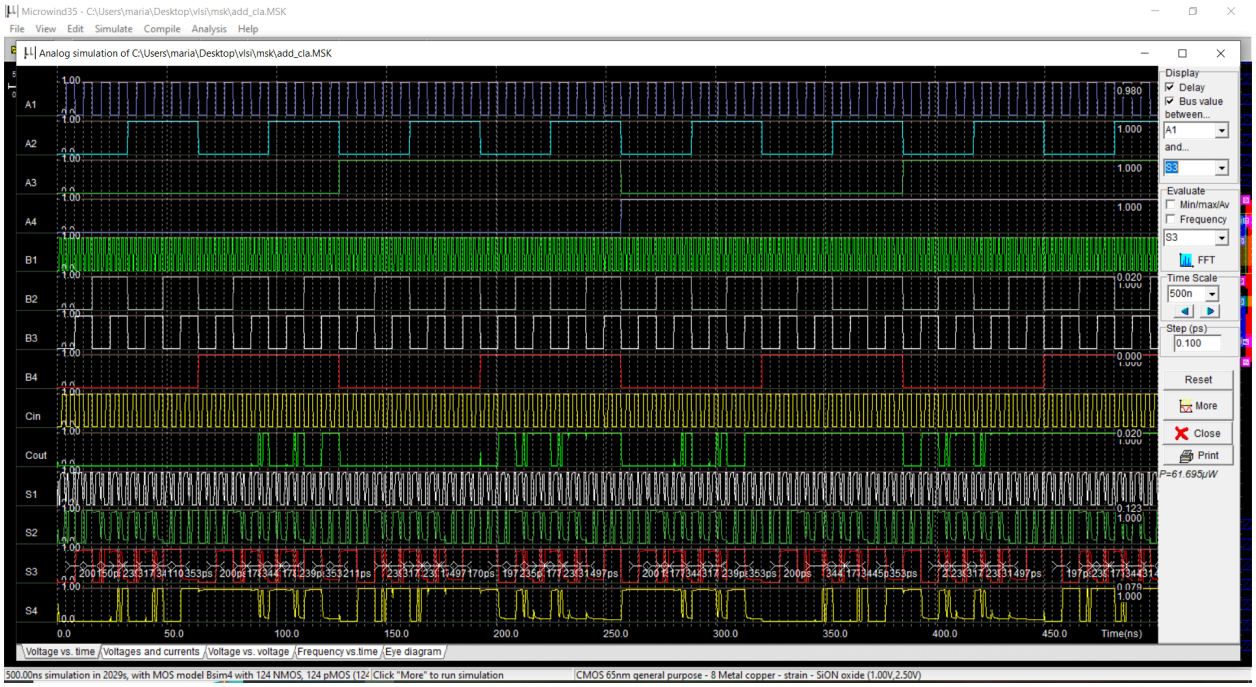
d. Cod Verilog

```
module add_cla_dsch( B1,Cin,A1,B3,B2,A2,B4,A3,
 A4,S1,S4,Cout,S2,S3);
 input B1,Cin,A1,B3,B2,A2,B4,A3;
 input A4;
 output S1,S4,Cout,S2,S3;
 wire w2,w3,w6,w7,w10,w11,w12,w16;
 wire w19,w21,w22,w23,w24,w25,w26,w27;
 wire w29,w30,w31,w32,w33,w34,w35,w36;
 wire w37,w38,w40,w41,w42,w43,w44,w45;
 wire w46,w47,w48,w49,w50,w51,w52,w55;
 wire w56,w57,w58;
 not #(1) inv_1(w3,w2);
 nor #(5) nor2_2(w6,B2,A2);
 nand #(5) nand2_3(w7,A2,B2);
 and #(3) and2_4(w12,w10,w11);
 nand #(4) nand2_5(w16,A4,B4);
 not #(1) inv_6(Cout,w19);
 not #(1) inv_7(w22,w21);
 and #(3) and2_8(w23,w22,w16);
 nor #(4) nor2_9(w21,B4,A4);
 nor #(2) nor3_10(w26,w6,w24,w25);
 xor #(3) xor2_11(S4,w23,w27);
 xor #(3) xor2_12(S1,w29,w3);
 and #(3) and2_13(w29,w30,w31);
 not #(1) inv_14(w30,w32);
 and #(3) and2_15(w33,w2,w31);
 or #(3) or2_16(w36,w34,w35);
 xor #(3) xor2_17(S2,w37,w38);
 and #(3) and3_18(w41,w32,w40,w7);
 and #(3) and2_19(w37,w42,w7);
 not #(1) inv_20(w42,w6);
 nor #(5) nor2_21(w32,B1,A1);
 nand #(4) nand2_22(w31,A1,B1);
 and #(3) and2_23(w45,w43,w44);
 and #(3) and2_24(w10,w2,w31);
 and #(3) and2_25(w46,w40,w6);
 and #(3) and3_26(w47,w6,w16,w40);
 and #(3) and2_27(w11,w7,w40);
 nor #(3) nor2_28(w38,w32,w33);
 or #(3) or2_29(w35,w41,w12);
 or #(3) or2_30(w19,w48,w49);
 not #(3) inv_31(w2,Cin);
 not #(1) inv_32(w51,w50);
 and #(3) and2_33(w52,w51,w40);
 xor #(3) xor2_34(S3,w52,w26);
 nand #(5) nand2_35(w40,A3,B3);
 nor #(4) nor2_36(w50,B3,A3);
 and #(3) and3_37(w25,w7,w31,w2);
 and #(3) and2_38(w24,w7,w32);
 or #(4) or3_39(w49,w47,w55,w45);
 and #(3) and2_40(w44,w40,w16);
 and #(3) and2_41(w55,w56,w57);
 and #(3) and2_42(w56,w7,w40);
 and #(3) and3_43(w43,w7,w31,w2);
 and #(3) and2_44(w57,w16,w32);
 and #(3) and2_45(w58,w16,w50);
 or #(3) or2_46(w34,w50,w46);
 or #(3) or2_47(w48,w21,w58);
 not #(1) inv_48(w27,w36);
endmodule

// Simulation parameters in Verilog Format
always
#200 B1=~B1;
#400 Cin=~Cin;
#800 A1=~A1;
#1600 B3=~B3;
#3200 B2=~B2;
#6400 A2=~A2;
#12800 B4=~B4;
#25600 A3=~A3;
#51200 A4=~A4;

// Simulation parameters
// B1 CLK 1 1
// Cin CLK 2 2
// A1 CLK 4 4
// B3 CLK 8 8
// B2 CLK 16 16
// A2 CLK 32 32
// B4 CLK 64 64
// A3 CLK 128 128
// A4 CLK 256 256
```

e. Implementare Microwind

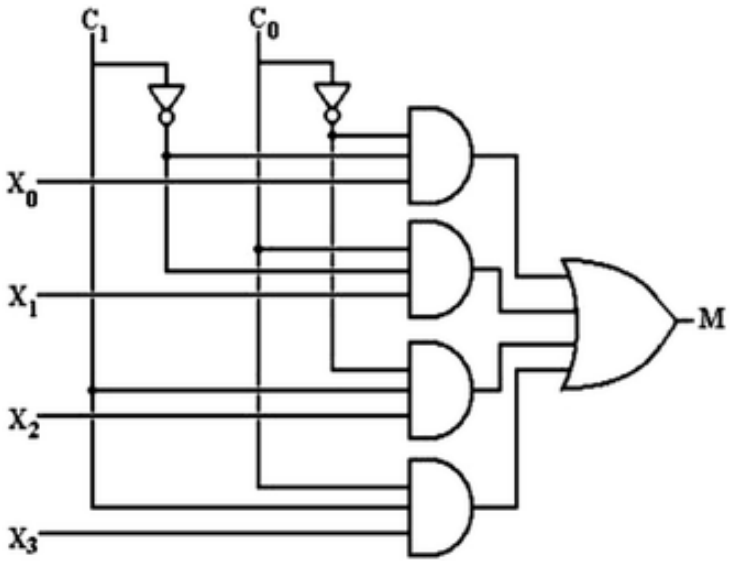


f. Simulare Microwind

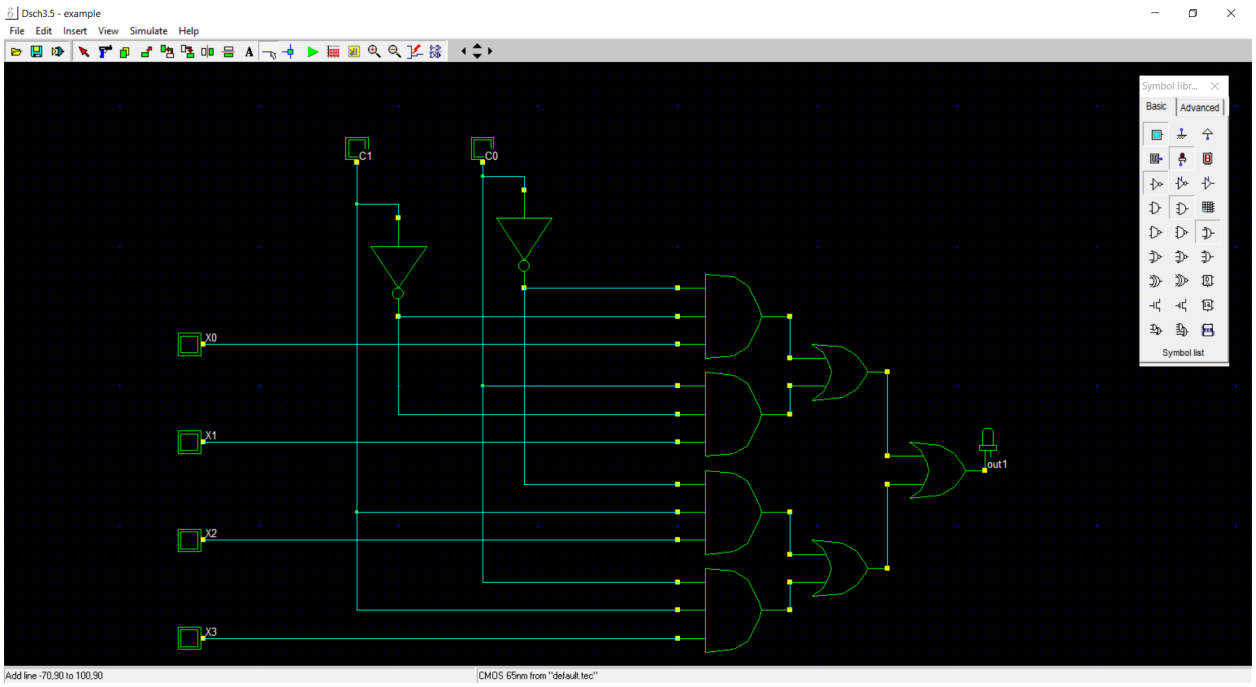


7. MUX 4:1

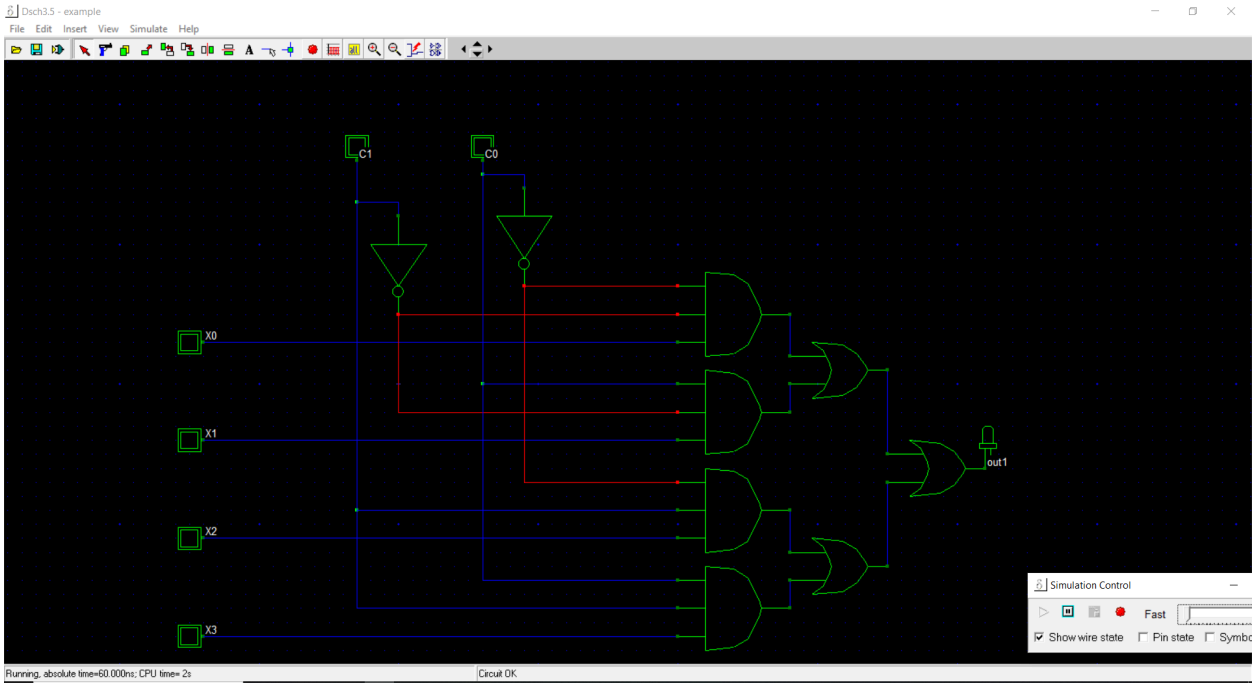
a. Schema circuitului

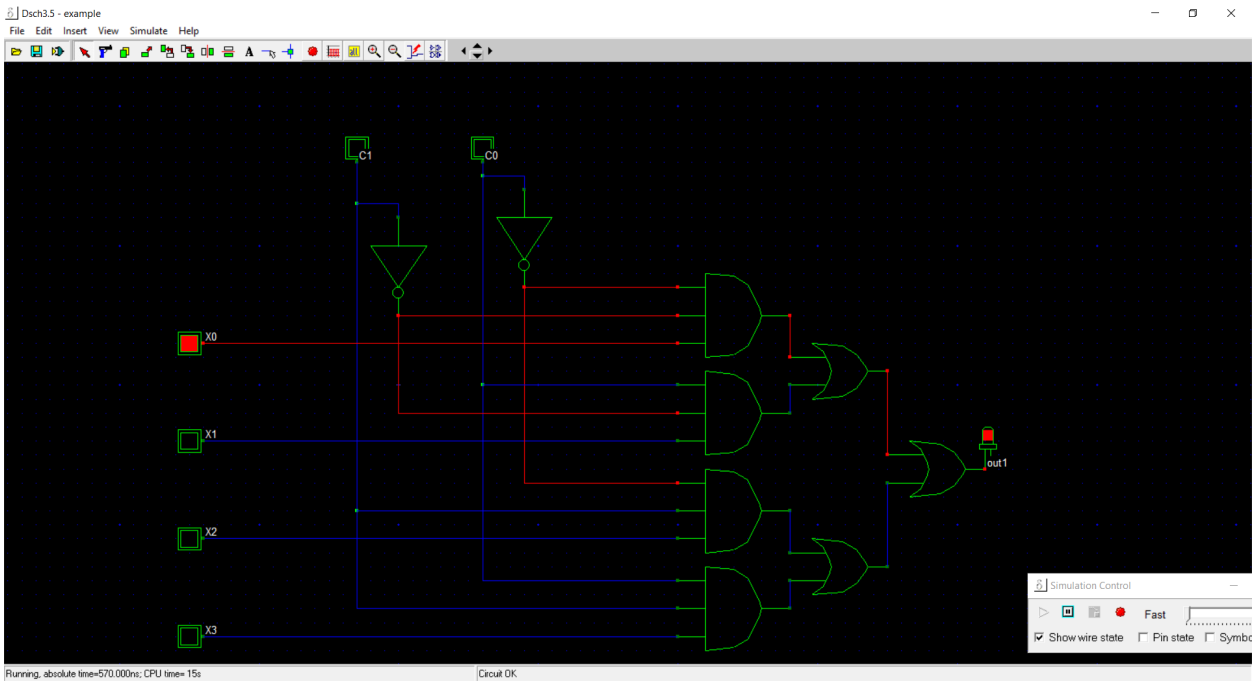


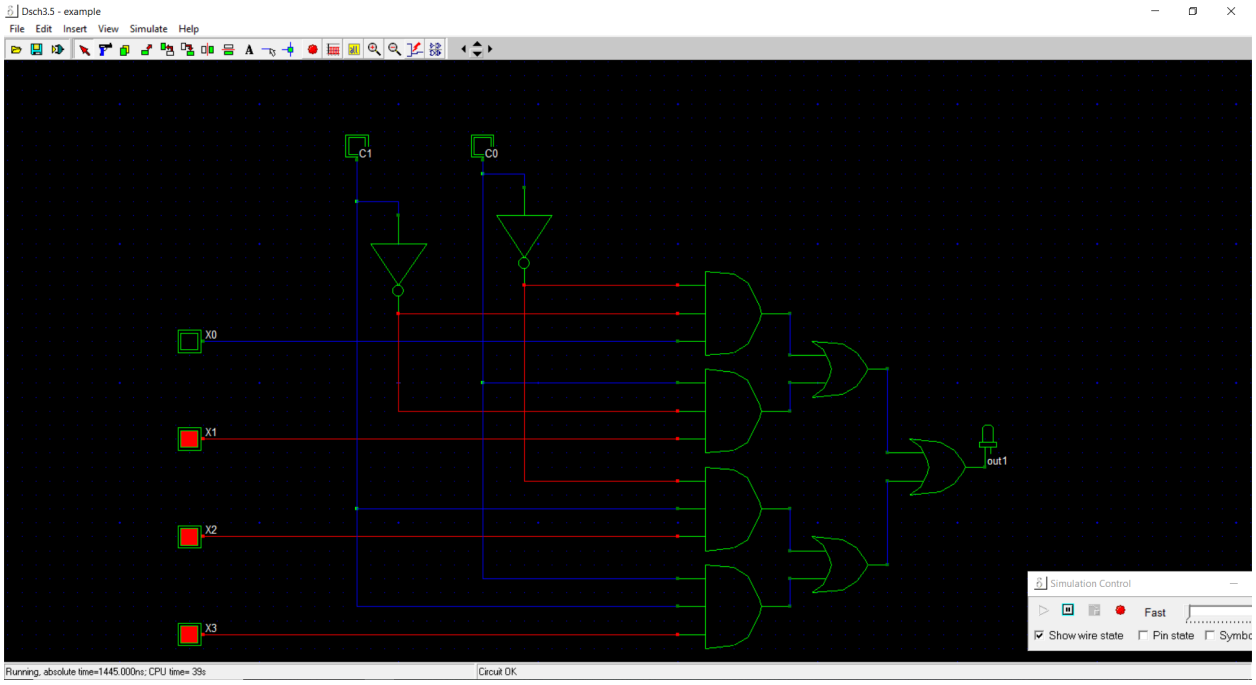
b. Implementare DSCH

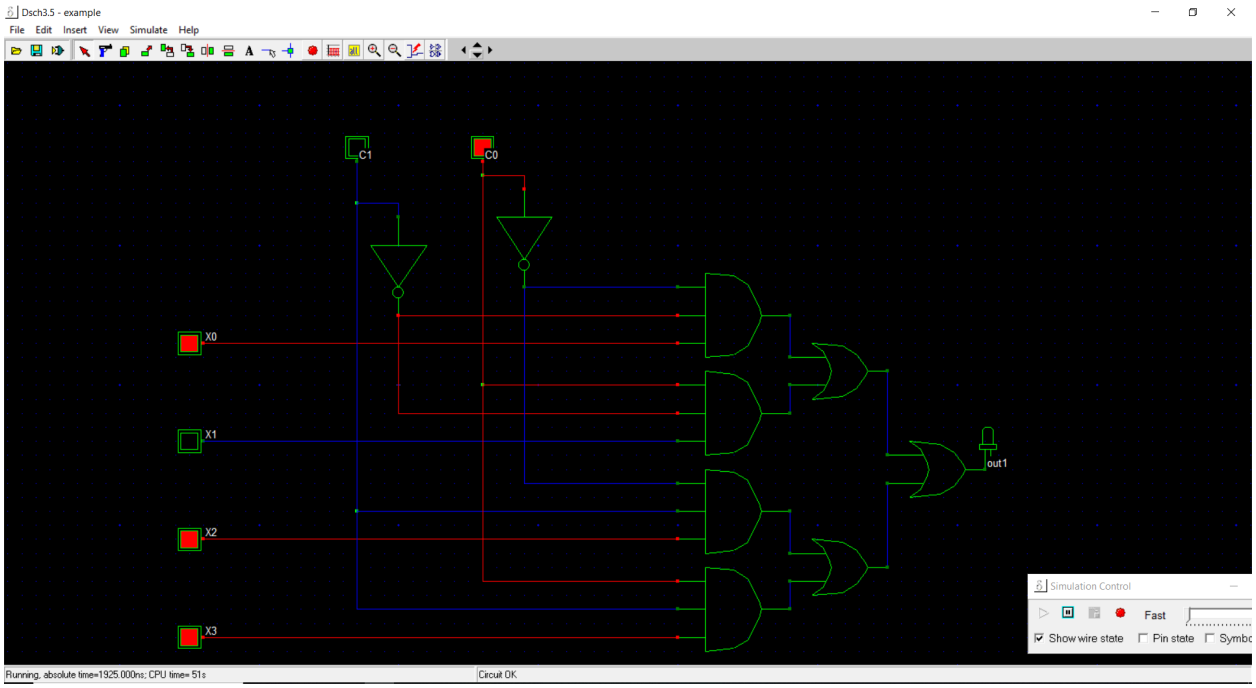


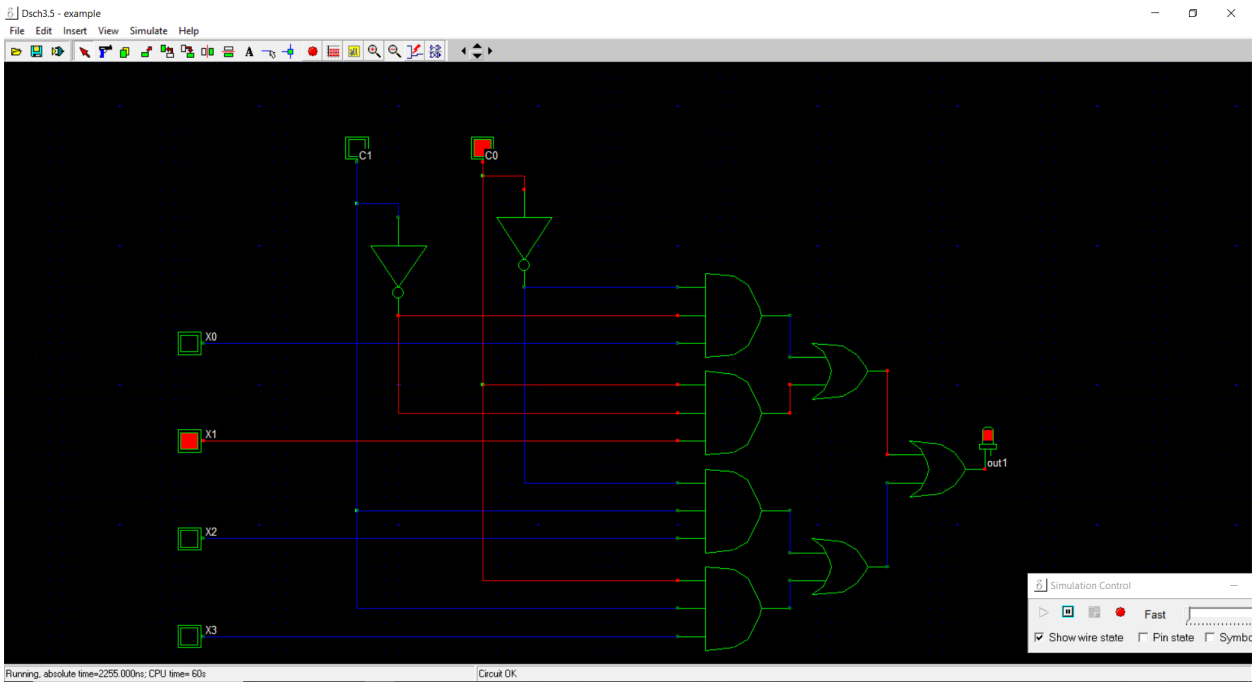
c. Simulare DSCH






Dsch3.5 - example
File Edit Insert View Simulate Help
C1
C0
X0
X1
X2
X3
out1
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=1445.000ns; CPU time= 39s
Circuit OK

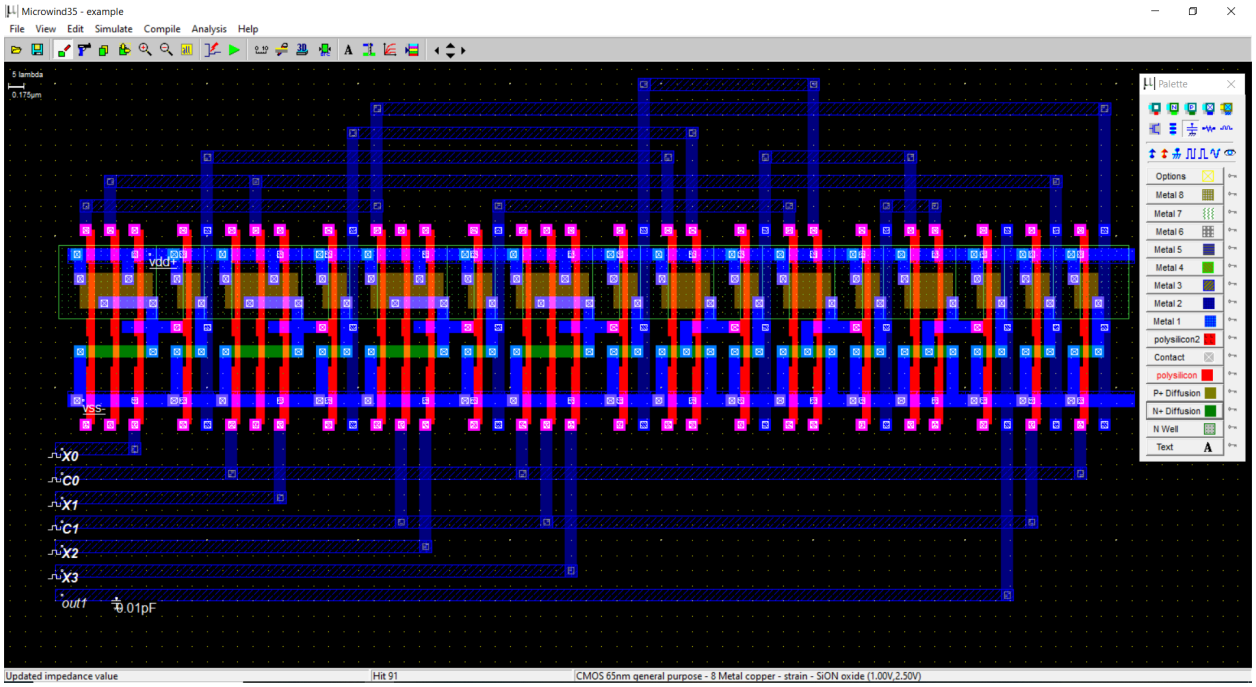

Dsch3.5 - example
File Edit Insert View Simulate Help
C1
C0
X0
X1
X2
X3
out1
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=1925.000ns; CPU time= 51s
Circuit OK



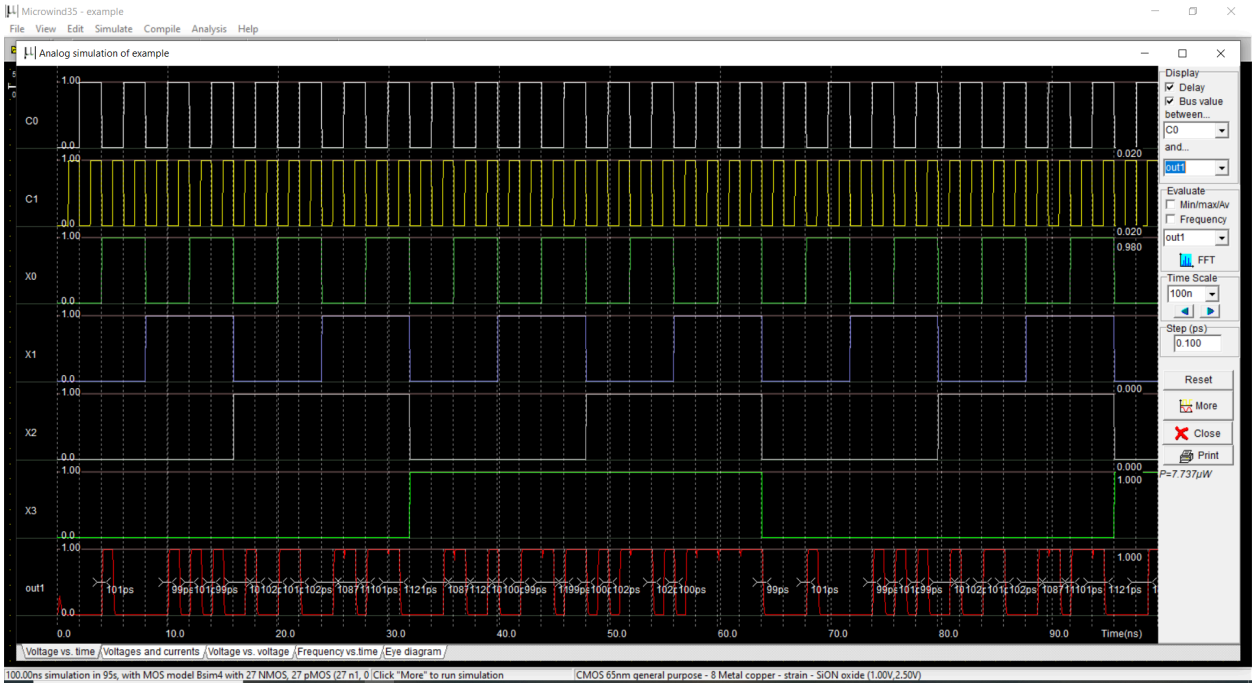
d. Cod Verilog

```
module mux_dsch( C1,C0,X0,X1,X2,X3,out1);
 input C1,C0,X0,X1,X2,X3;
 output out1;
 wire w2,w3,w5,w8,w11,w13,w14,w15;
 wire ;
 and #(3) and3_1(w5,w2,w3,X0);
 and #(3) and3_2(w8,C0,w3,X1);
 and #(3) and3_3(w11,w2,C1,X2);
 and #(3) and3_4(w13,C0,C1,X3);
 or #(3) or2_5(w14,w5,w8);
 or #(3) or2_6(w15,w11,w13);
 or #(3) or2_7(out1,w14,w15);
 not #(2) inv_8(w3,C1);
 not #(2) inv_9(w2,C0);
endmodule

// Simulation parameters in Verilog Format
always
#200 C1=~C1;
#400 C0=~C0;
#800 X0=~X0;
#1600 X1=~X1;
#3200 X2=~X2;
#6400 X3=~X3;

// Simulation parameters
// C1 CLK 1 1
// C0 CLK 2 2
// X0 CLK 4 4
// X1 CLK 8 8
// X2 CLK 16 16
// X3 CLK 32 32
```

e. Implementare Microwind

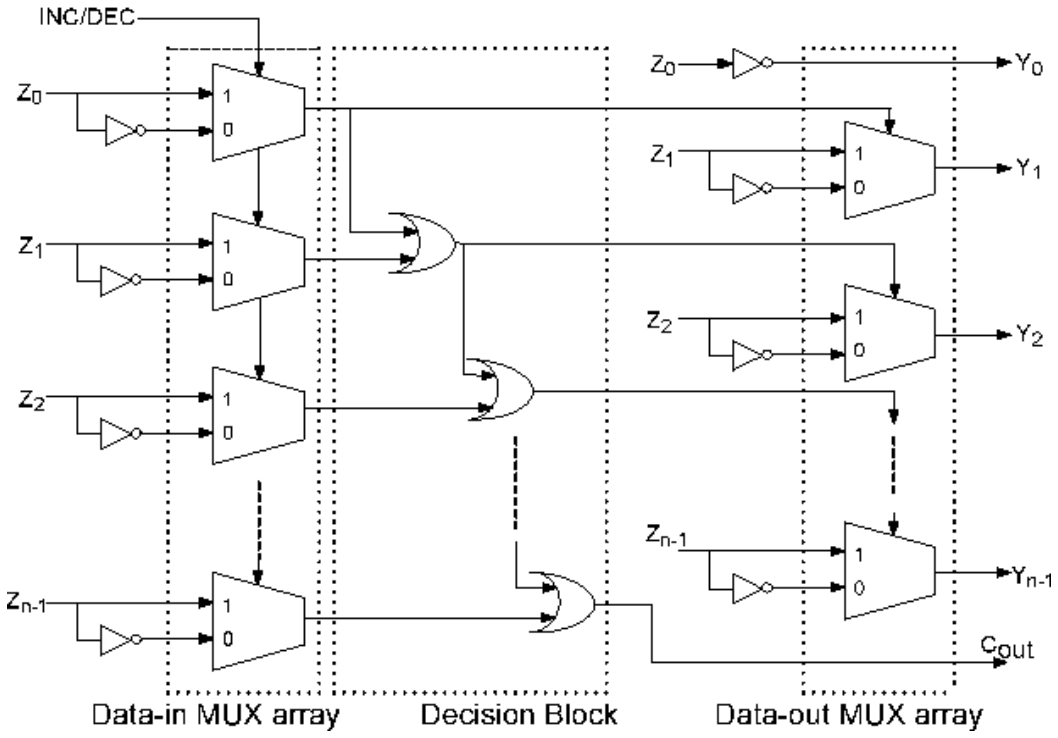


f. Simulare Microwind



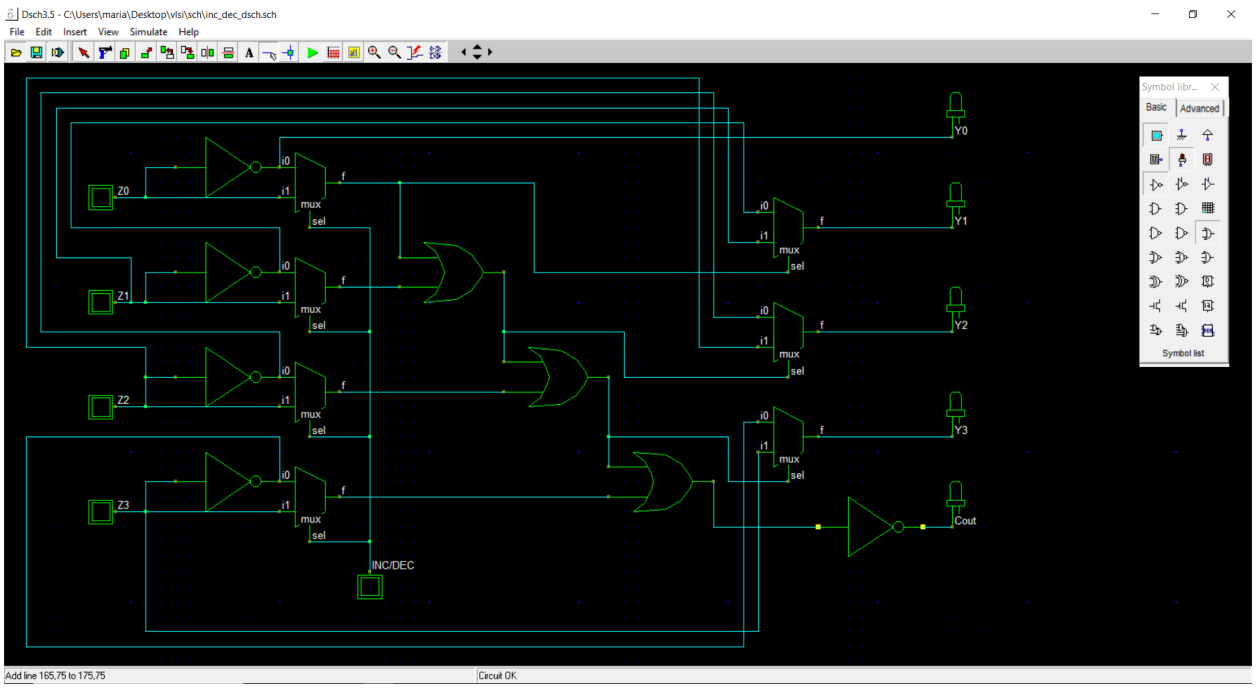
8. Incrementator / decrementator 4 biți

a. Schema circuitului

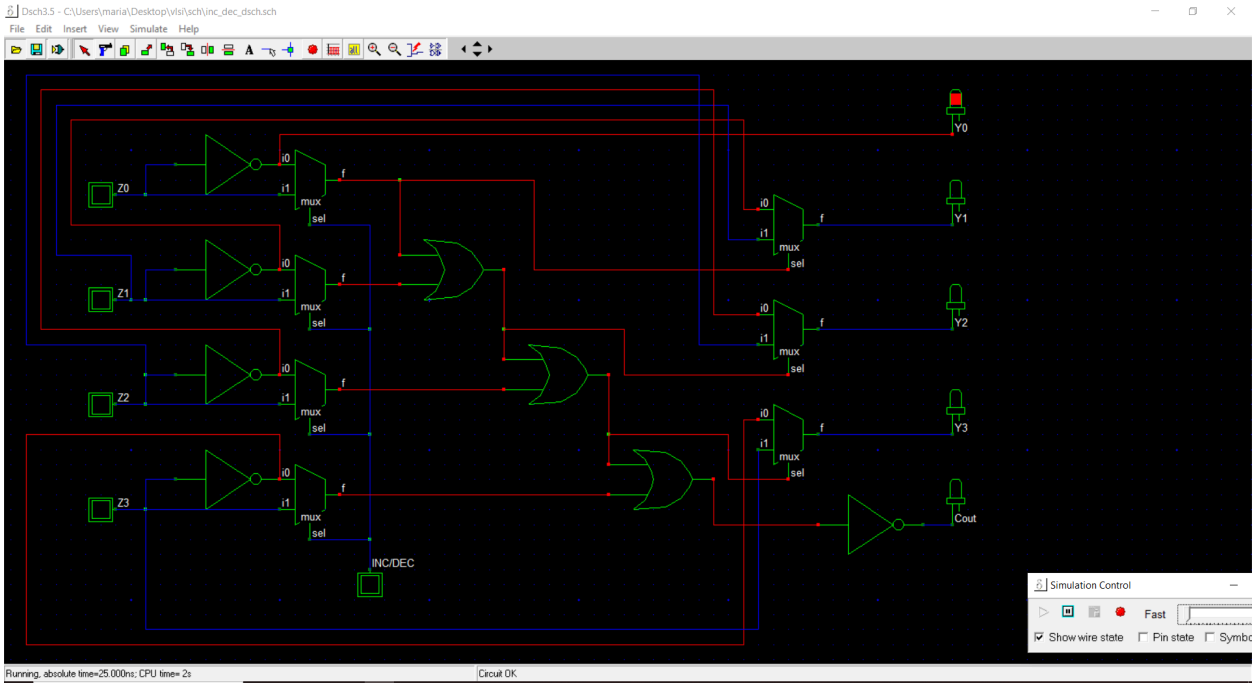


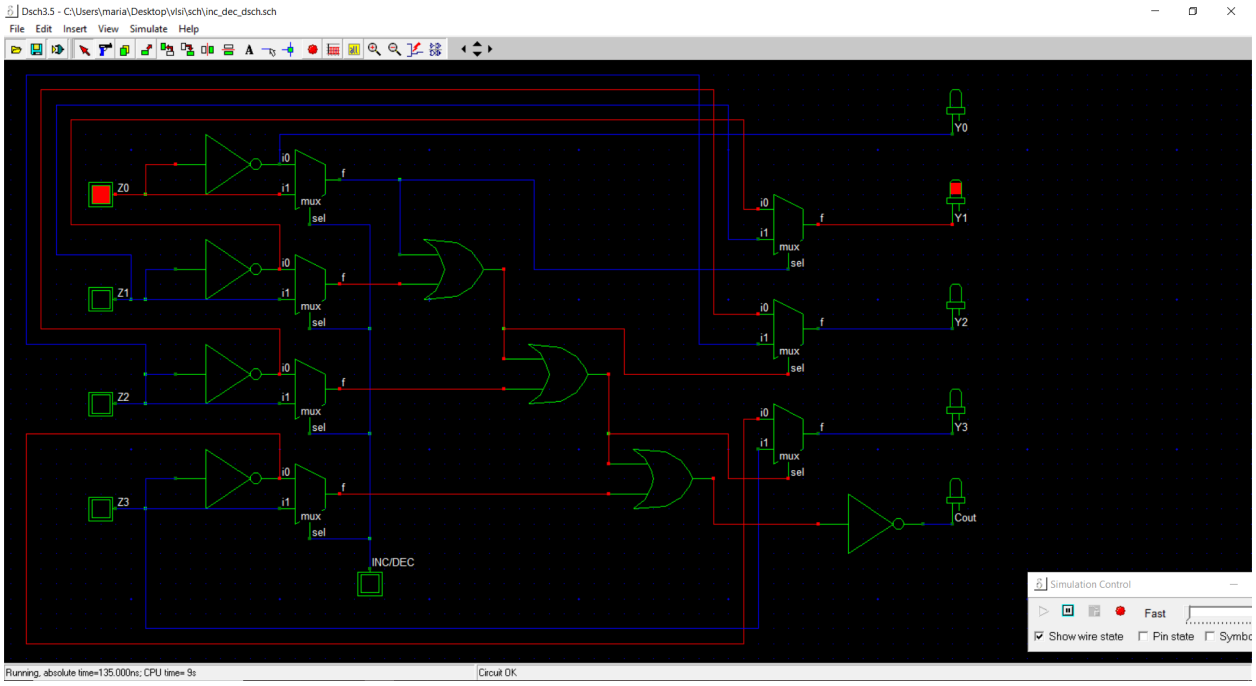
Sursa: https://www.semanticscholar.org/paper/Increment-decrement-2's-complement-priority-encoder-Phaneendra-Vudadha/8c38e9757bef60af55cd5c5ce8ab0d65b6fcfa1a/figure/0

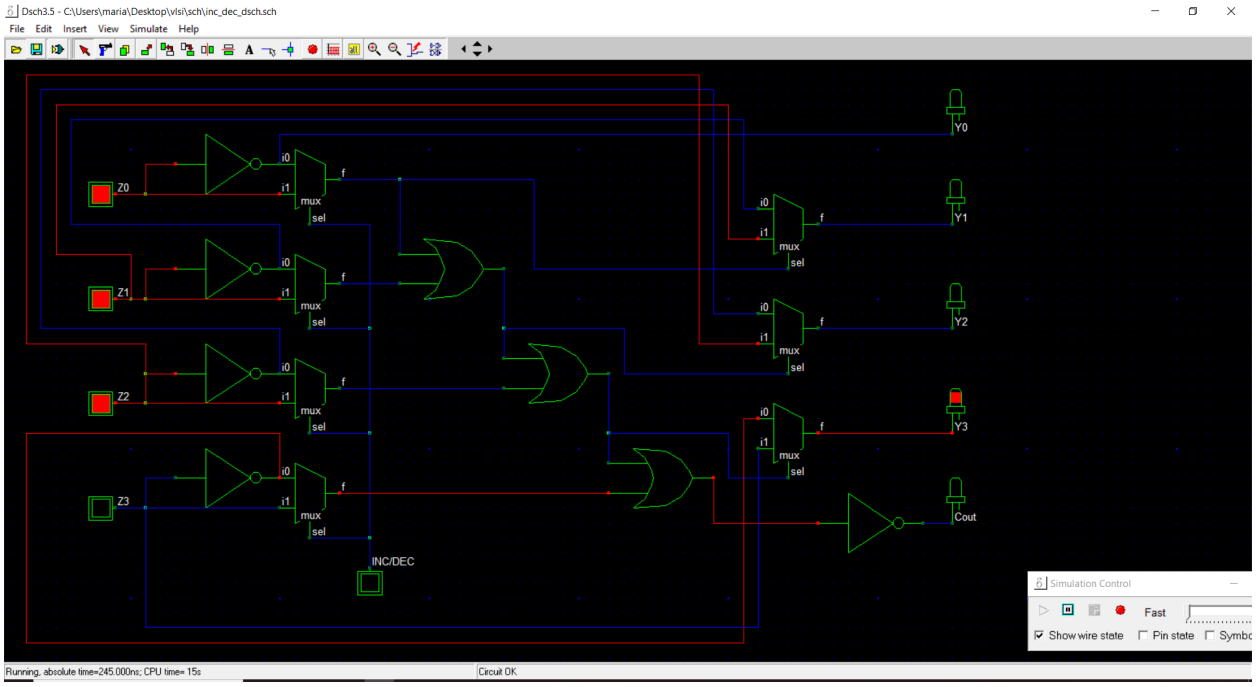
b. Implementare DSCH

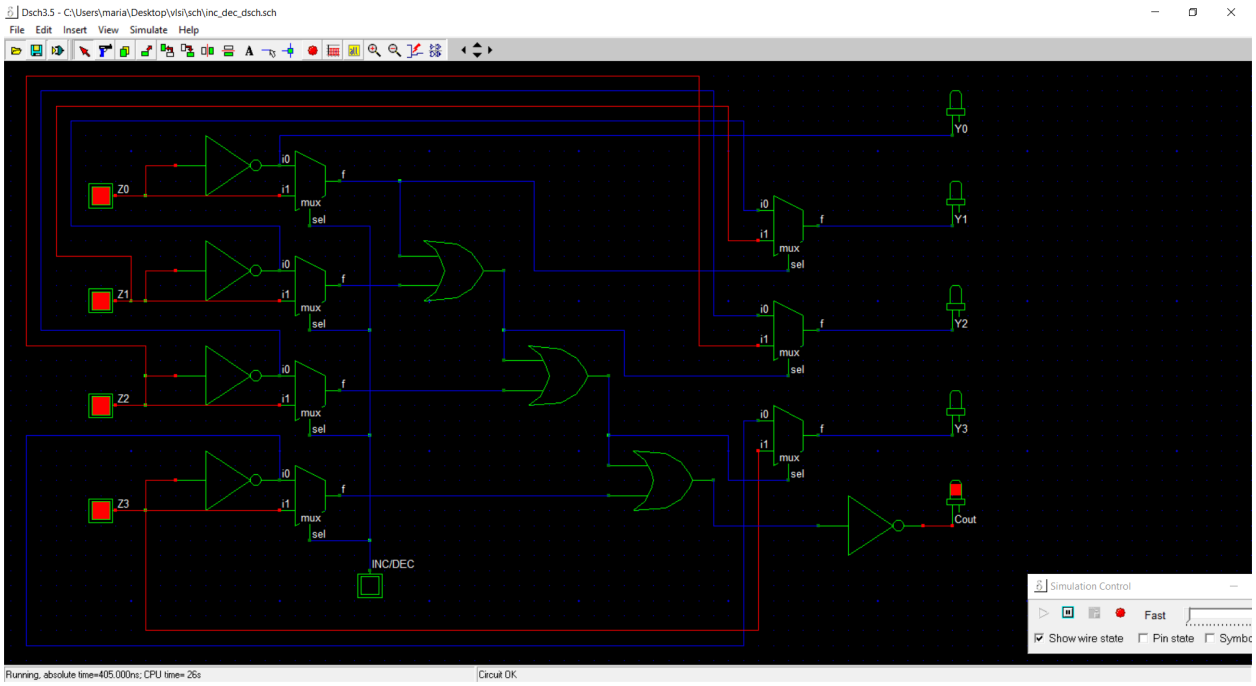


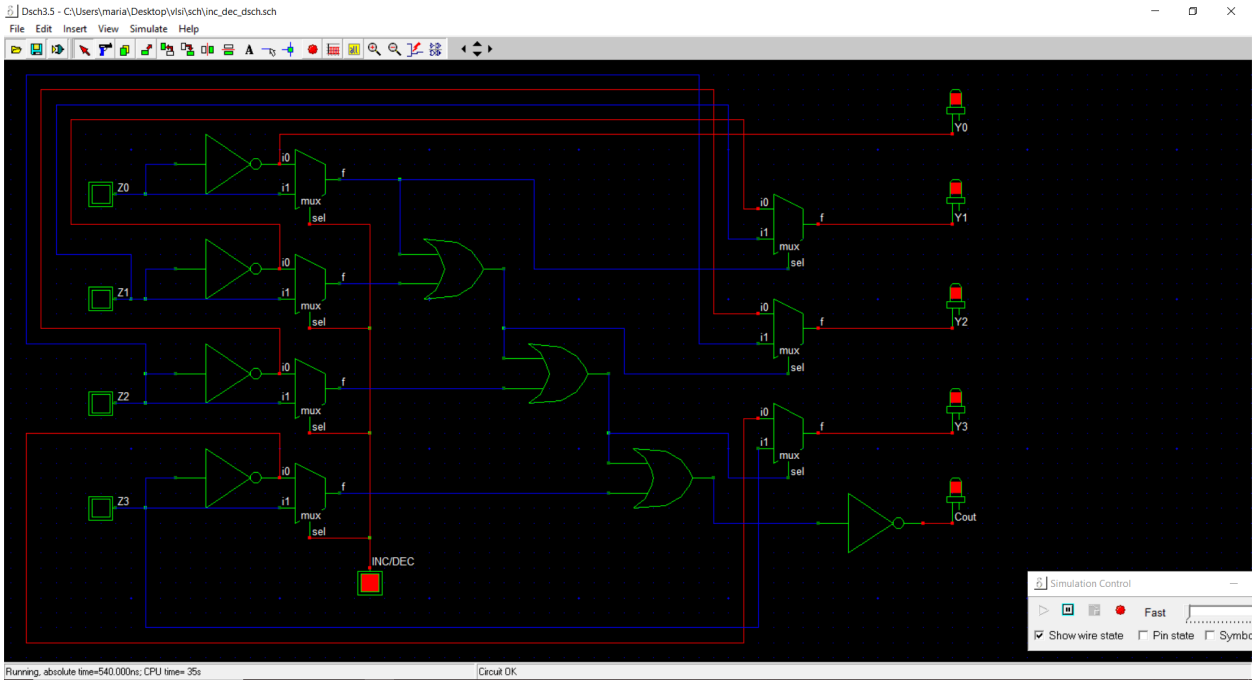
c. Simulare DSCH

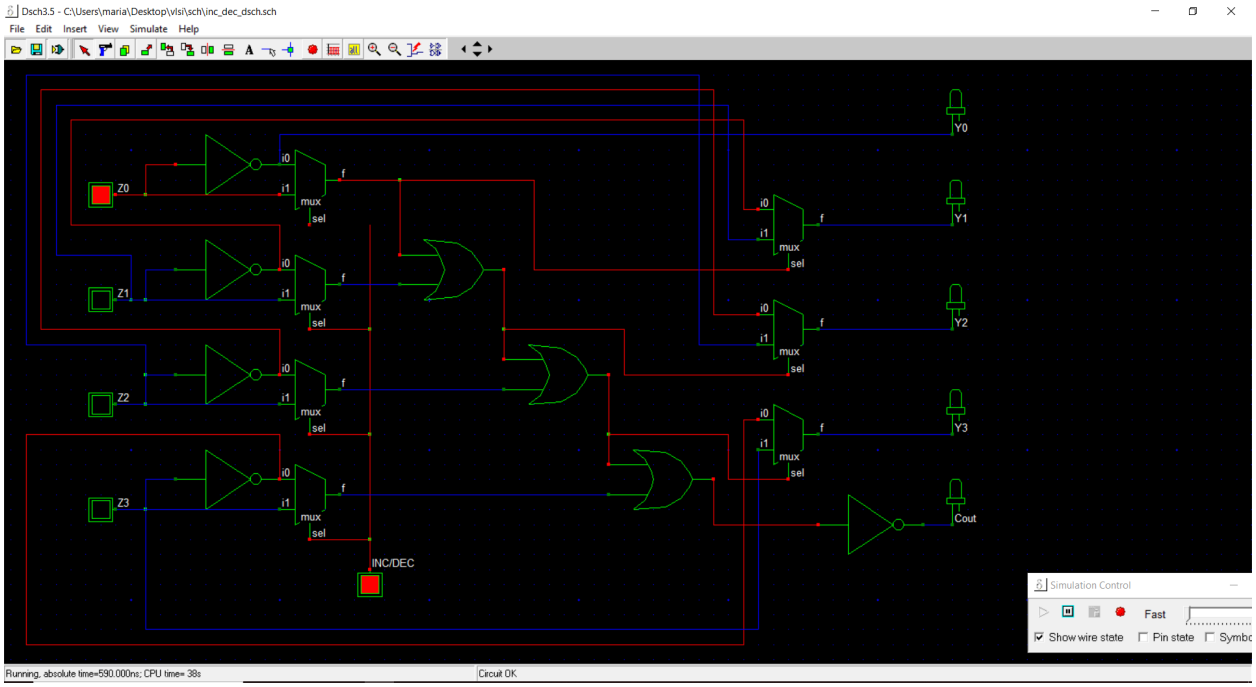


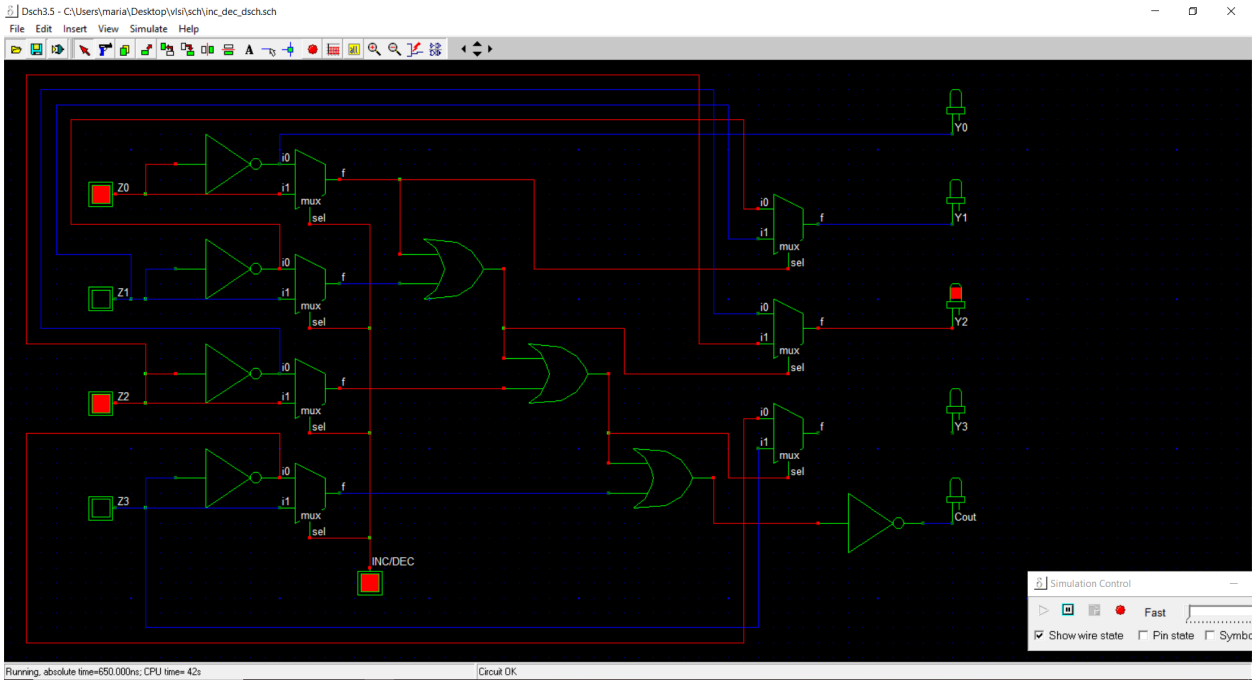



Dsch3.5 - C:\Users\maria\Desktop\visi\sch\inc_dec_dsch.sch
File Edit Insert View Simulate Help
Z0
Z1
Z2
Z3
i0
i1
mux
sel
f
Y0
Y1
Y2
Y3
Cout
INC/DEC
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=245.000ns; CPU time= 15s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\visi\sch\inc_dec_dsch.sch
File Edit Insert View Simulate Help
Z0
Z1
Z2
Z3
i0
i1
mux
sel
f
Y0
Y1
Y2
Y3
Cout
INC/DEC
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=405.000ns; CPU time= 26s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\inc_dec_dsch.sch
File Edit Insert View Simulate Help
Z0
Z1
Z2
Z3
INC/DEC
Y0
Y1
Y2
Y3
Cout
i0
i1
mux
sel
f
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=540.000ns; CPU time= 35s
Circuit OK

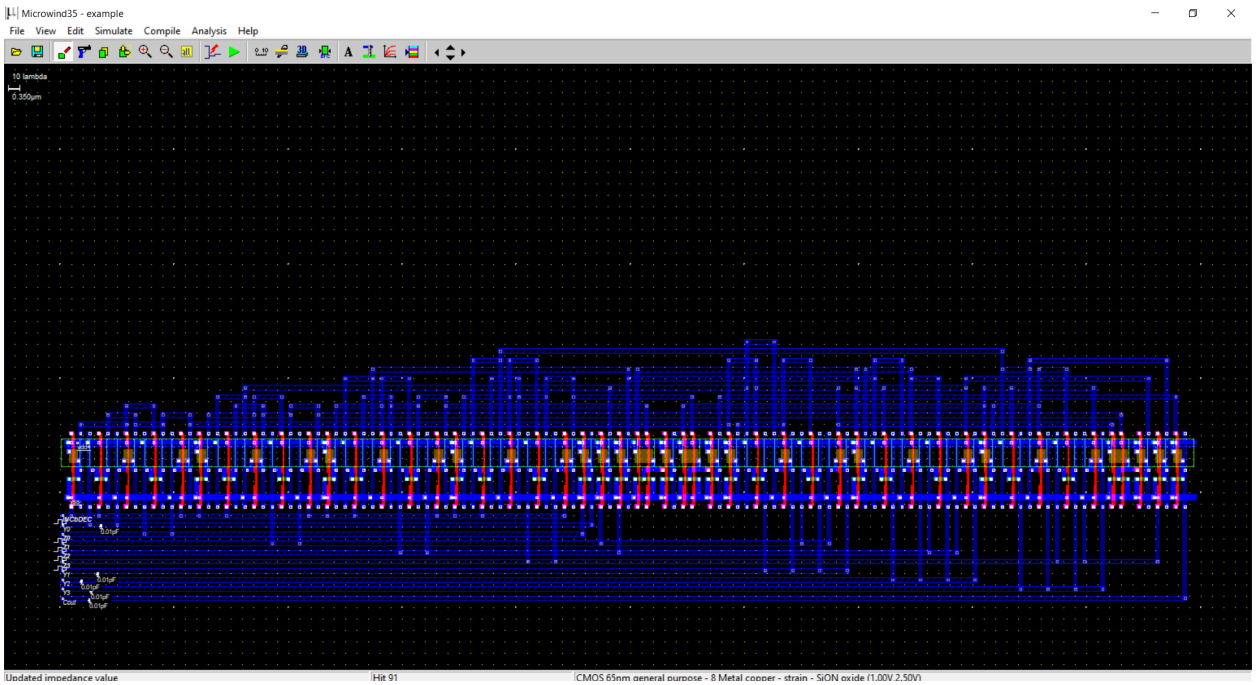

Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\inc_dec_dsch.sch
File Edit Insert View Simulate Help
Z0
Z1
Z2
Z3
INC/DEC
Y0
Y1
Y2
Y3
Cout
i0
i1
mux
sel
f
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=590.000ns; CPU time= 38s
Circuit OK



d. Cod Verilog

```
module inc_dec_dsch( INCbDEC,Z0,Z1,Z2,Z3,Y3,Y0,Y1,
 Y2,Cout);
 input INCbDEC,Z0,Z1,Z2,Z3;
 output Y3,Y0,Y1,Y2,Cout;
 wire w5,w6,w8,w9,w11,w12,w14,w16;
 wire w17,w18,;
 mux #(2) mux_1(w5,Y0,Z0,INCbDEC);
 mux #(1) mux_2(w8,w6,Z1,INCbDEC);
 mux #(1) mux_3(w11,w9,Z2,INCbDEC);
 mux #(1) mux_4(w14,w12,Z3,INCbDEC);
 not #(2) inv_5(Y0,Z0);
 not #(2) inv_6(w6,Z1);
 not #(2) inv_7(w9,Z2);
 or #(4) or2_8(w17,w16,w11);
 or #(3) or2_9(w18,w17,w14);
 mux #(1) mux_10(Y1,w6,Z1,w5);
 mux #(1) mux_11(Y2,w9,Z2,w16);
 mux #(1) mux_12(Y3,w12,Z3,w17);
 or #(4) or2_13(w16,w5,w8);
 not #(2) inv_14(w12,Z3);
 not #(1) inv_15(Cout,w18);
endmodule

// Simulation parameters in Verilog Format
always
#200 INC/DEC=~INC/DEC;
#400 Z0=~Z0;
#800 Z1=~Z1;
#1600 Z2=~Z2;
#3200 Z3=~Z3;

// Simulation parameters
// INC/DEC CLK 1 1
// Z0 CLK 2 2
// Z1 CLK 4 4
// Z2 CLK 8 8
// Z3 CLK 16 16
```

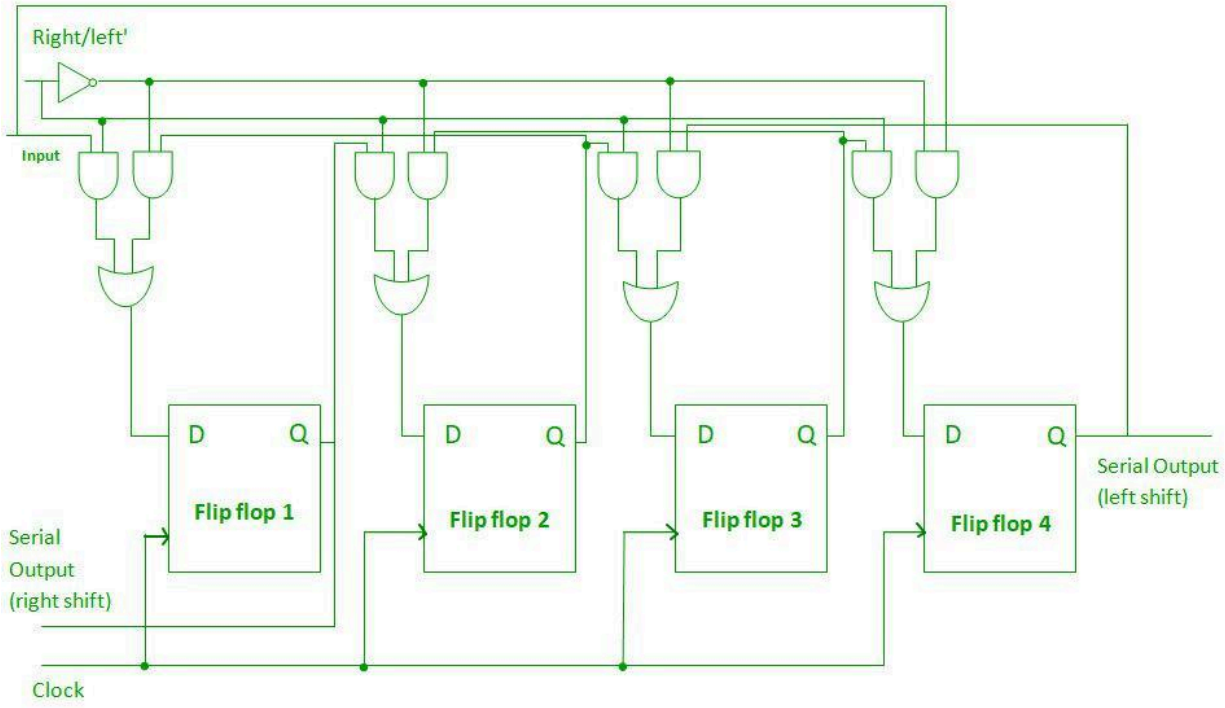
e. Implementare Microwind



f. Simulare Microwind

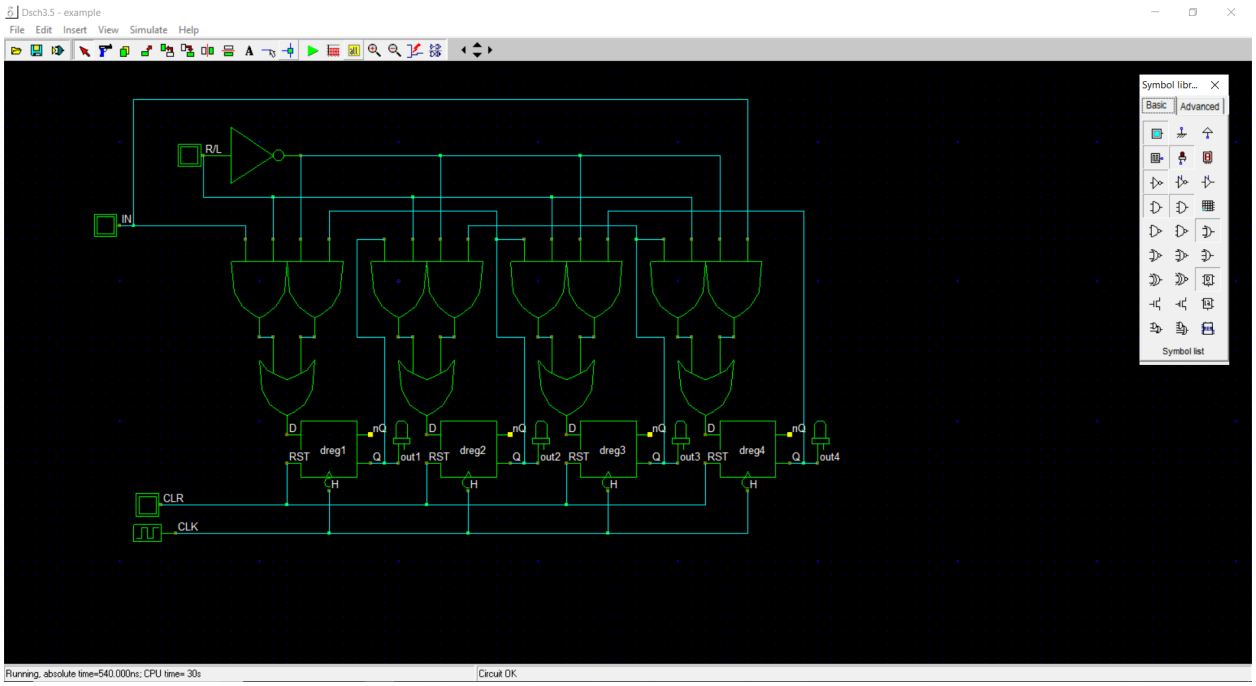
9. Registru deplasare stânga/dreapta 4 biți

a. Schema circuitului

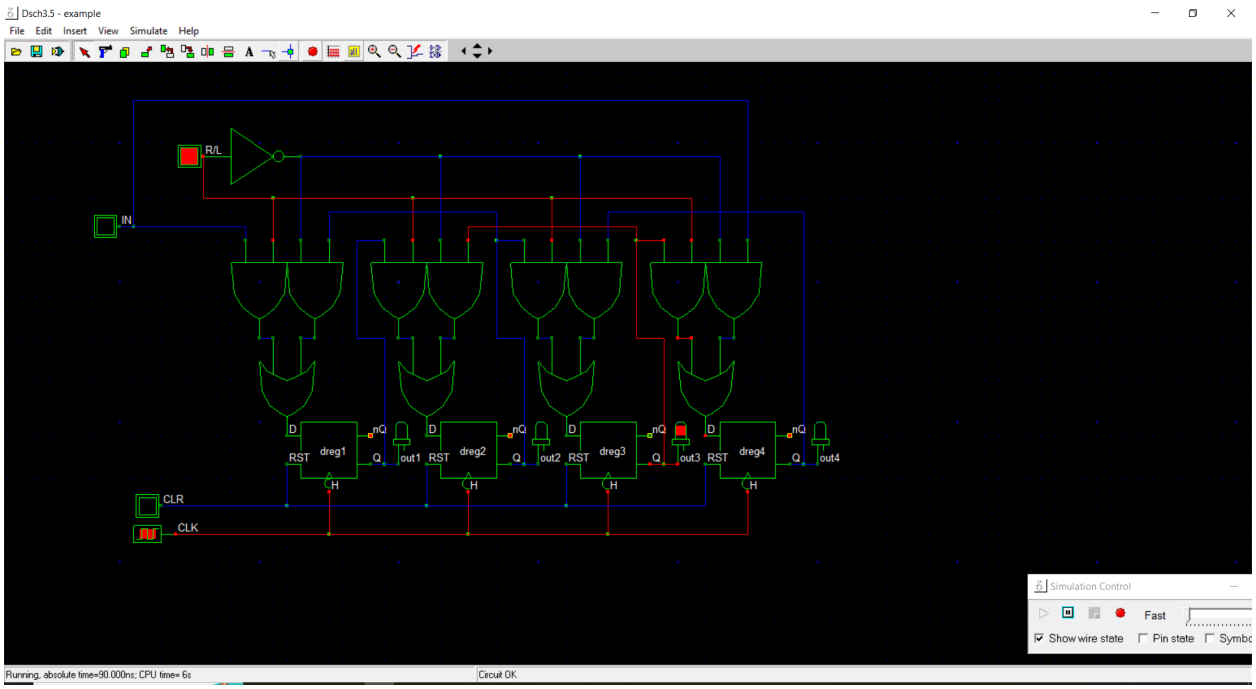


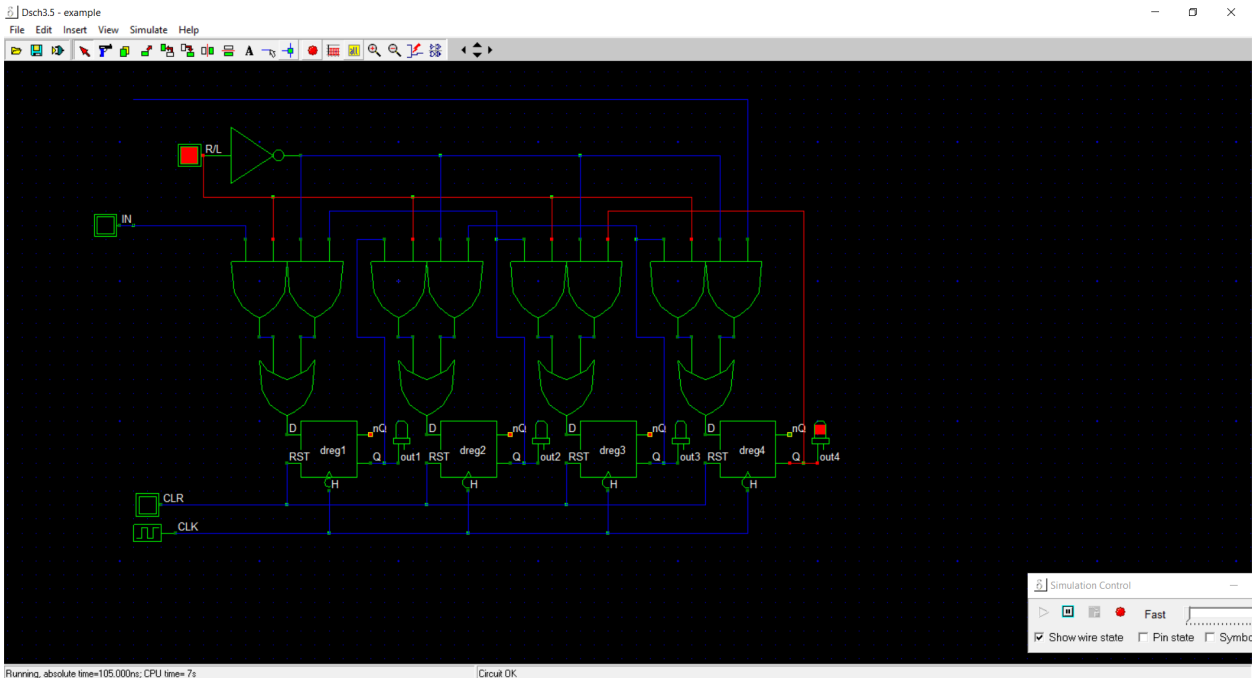
Sursa: https://www.geeksforgeeks.org/shift-registers-in-digital-logic/

b. Implementare DSCH



c. Simulare DSCH

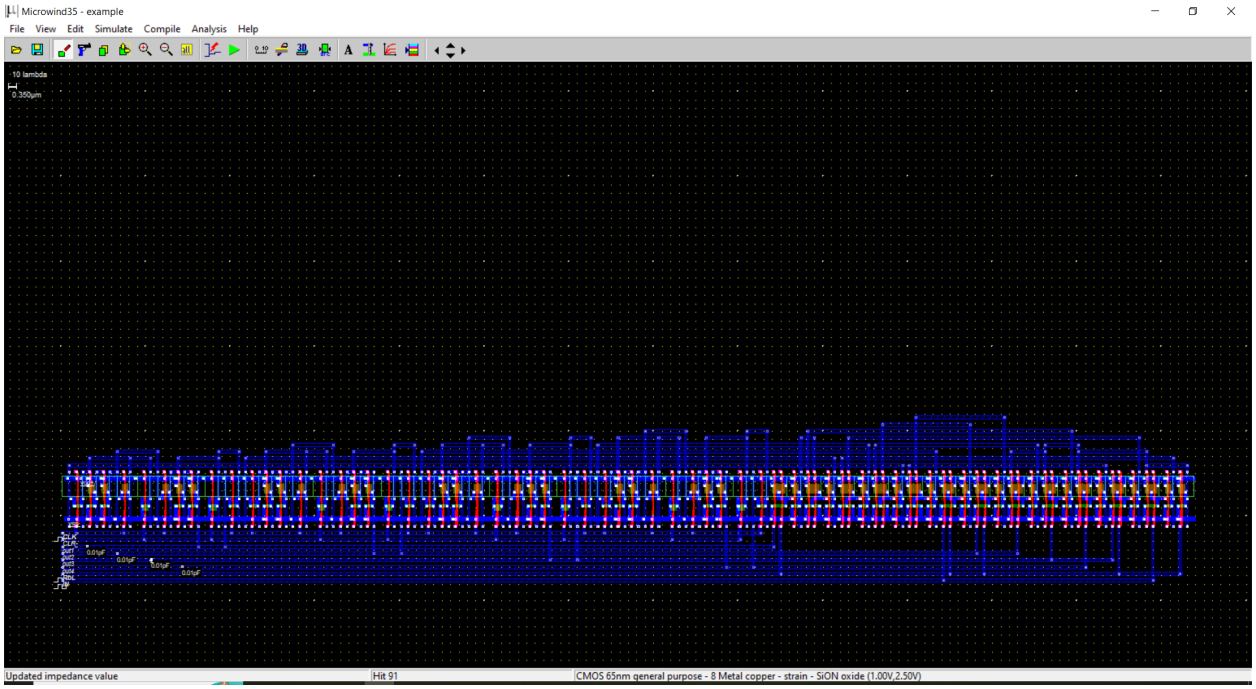




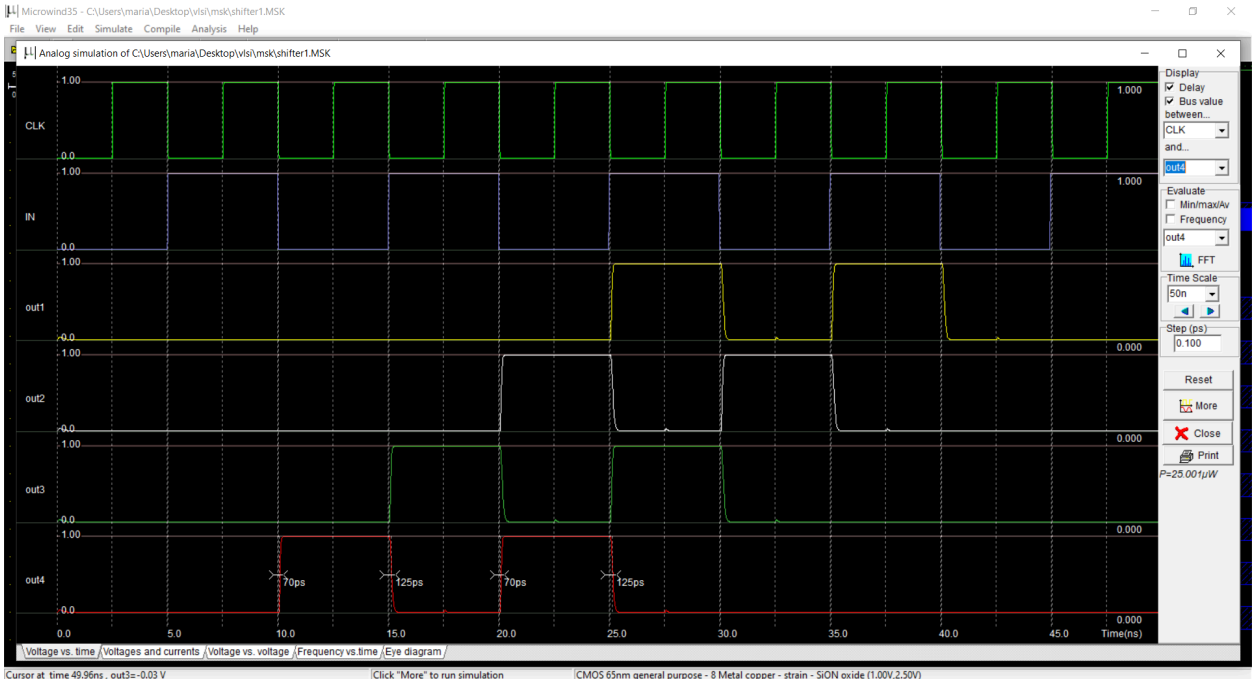
d. Cod Verilog

```
module shifter1_dsch( CLK,CLR,RbL,IN,out1,out2,out3,out4);
 input CLK,CLR,RbL,IN;
 output out1,out2,out3,out4;
 wire w2,w6,w7,w9,w10,w12,w13,w15;
 wire w17,w18,w19,w20,w21,w22,w23,w24;
 wire w26;
 dreg #(4) dreg_1(out1,w6,w2,CLR,CLK);
 dreg #(4) dreg_2(out2,w9,w7,CLR,CLK);
 dreg #(4) dreg_3(out3,w12,w10,CLR,CLK);
 dreg #(4) dreg_4(out4,w15,w13,CLR,CLK);
 and #(3) and2_5(w17,out1,RbL);
 or #(3) or2_6(w7,w18,w17);
 or #(3) or2_7(w13,w19,w20);
 or #(3) or2_8(w2,w21,w22);
 or #(3) or2_9(w10,w23,w24);
 and #(3) and2_10(w22,IN,RbL);
 and #(3) and2_11(w24,out2,RbL);
 and #(3) and2_12(w21,w26,out2);
 and #(3) and2_13(w18,w26,out3);
 and #(3) and2_14(w23,w26,out4);
 and #(3) and2_15(w20,out3,RbL);
 and #(3) and2_16(w19,w26,IN);
 not #(3) inv_17(w26,RbL);
endmodule

// Simulation parameters in Verilog Format
always
#2000 CLK=~CLK;
#200 CLR=~CLR;
#400 R/L=~R/L;
#800 IN=~IN;

// Simulation parameters
// CLK CLK 10 10
// CLR CLK 1 1
// R/L CLK 2 2
// IN CLK 4 4
```

e. Implementare Microwind

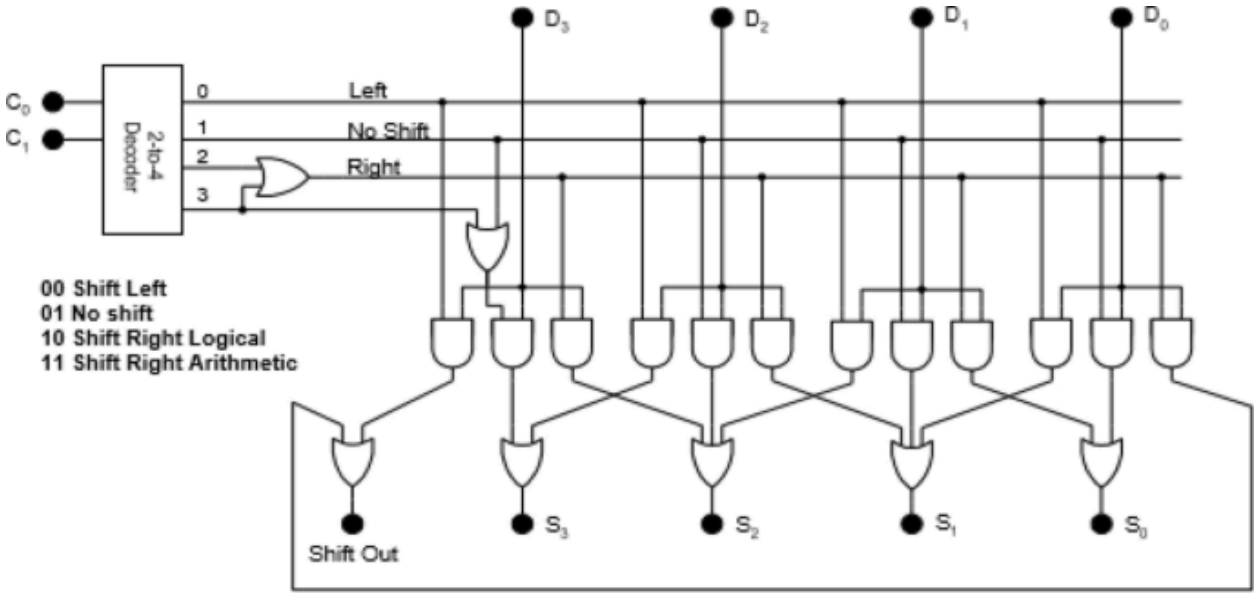


f. Simulare Microwind



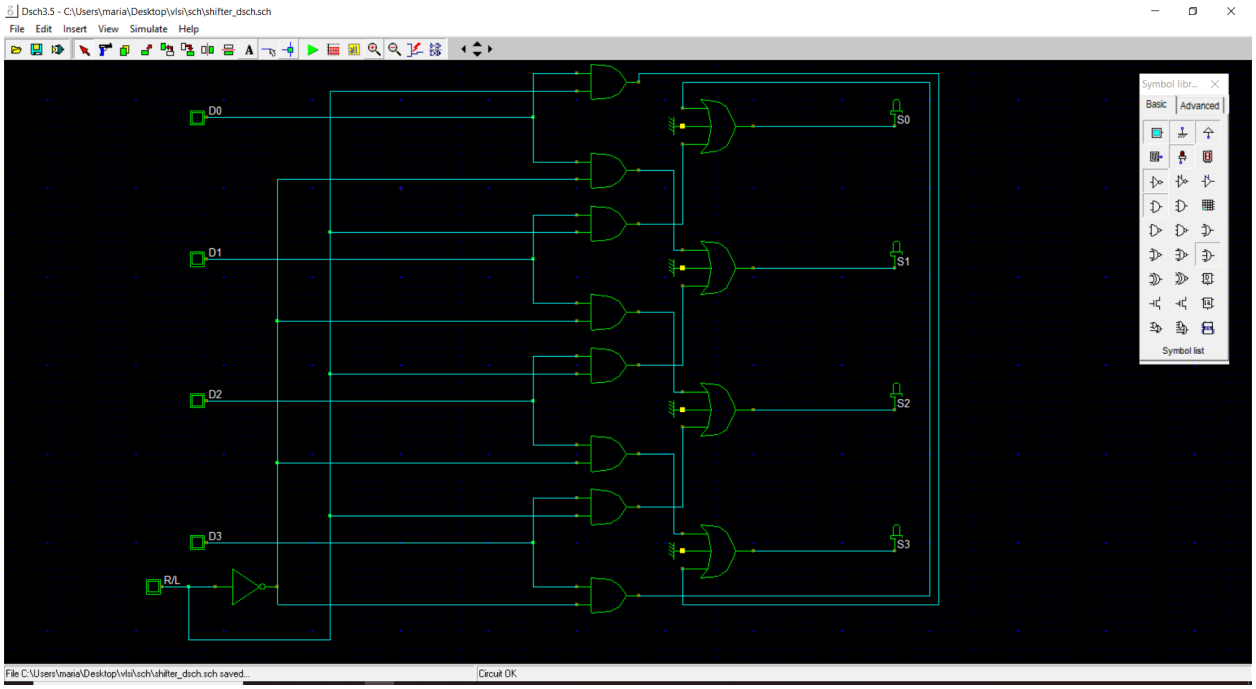
9’. Registru deplasare stânga/dreapta 4 biți

a. Schema circuitului

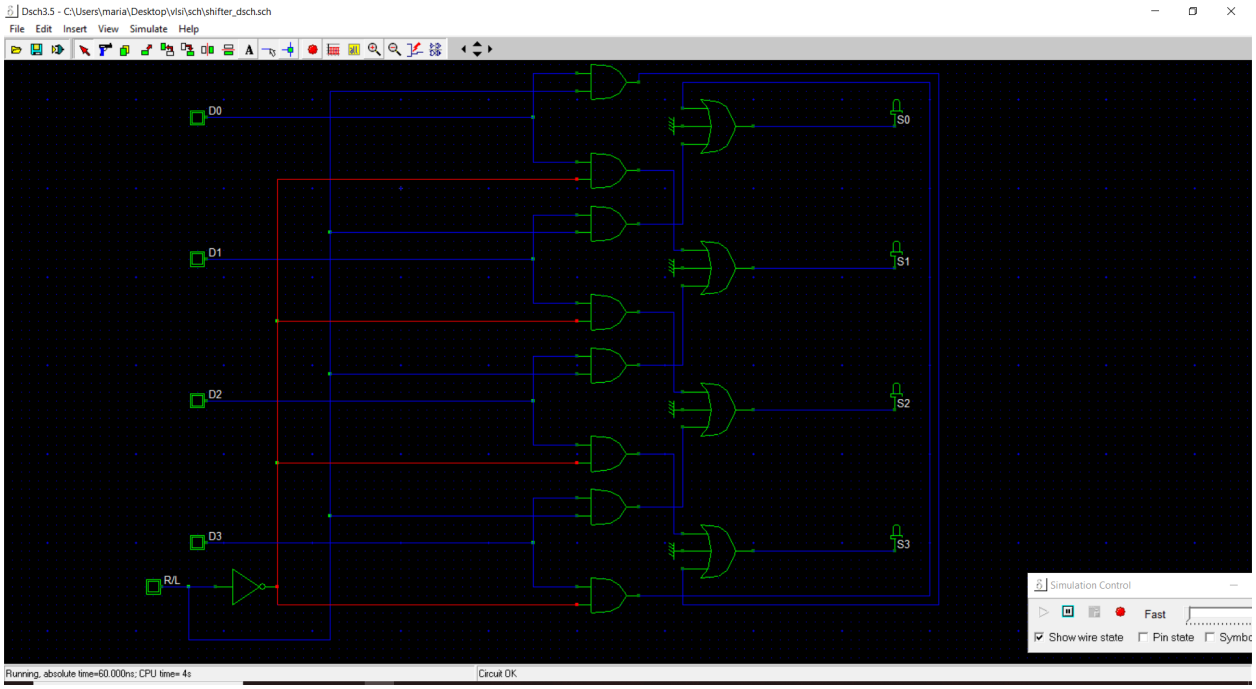


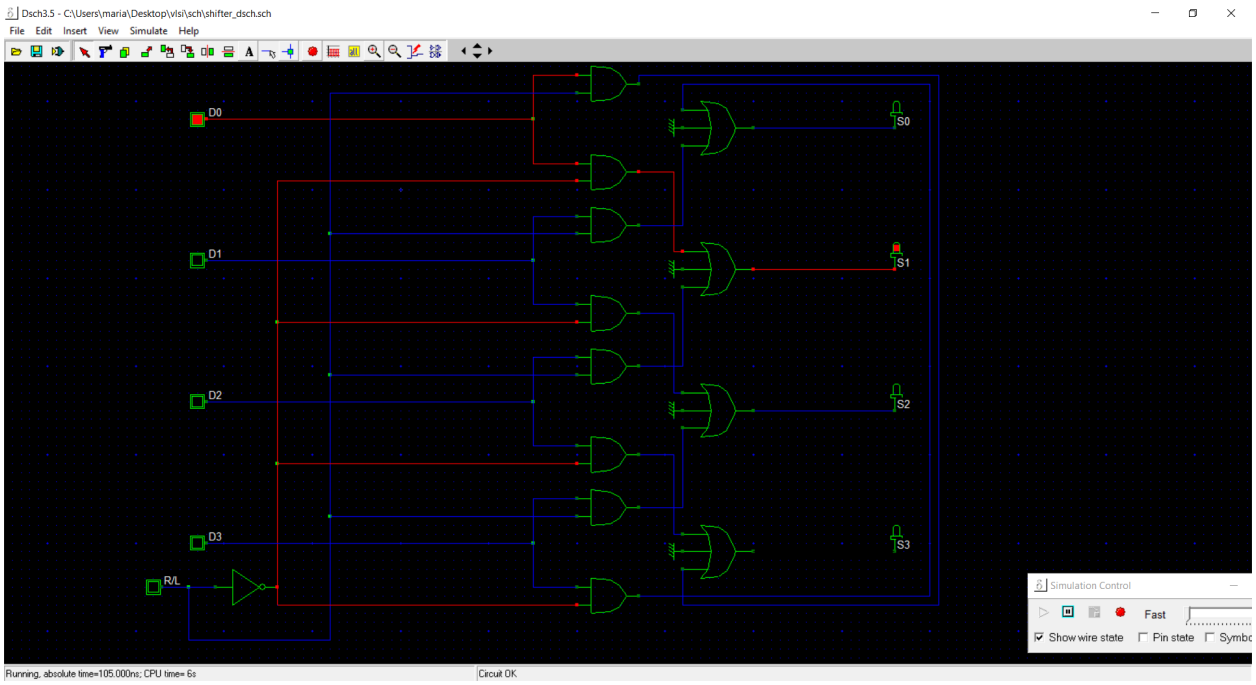
Sursa: https://jesit.springeropen.com/articles/10.1186/s43067-021-00029-8

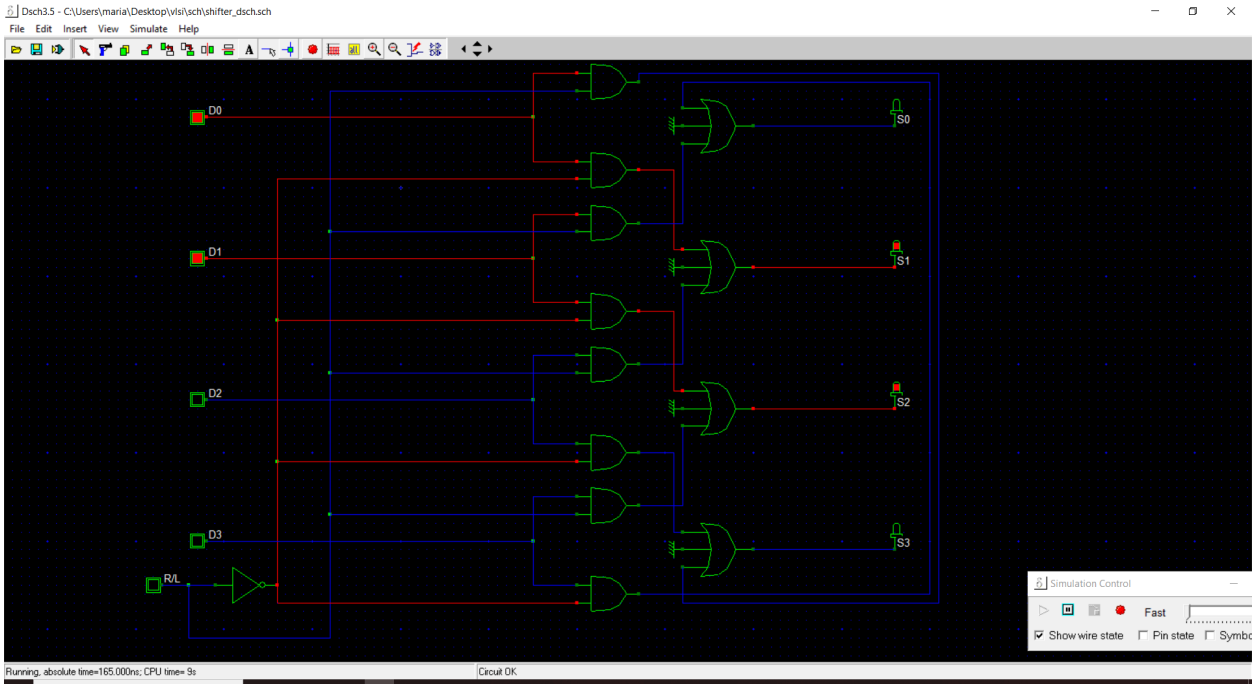
b. Implementare DSCH

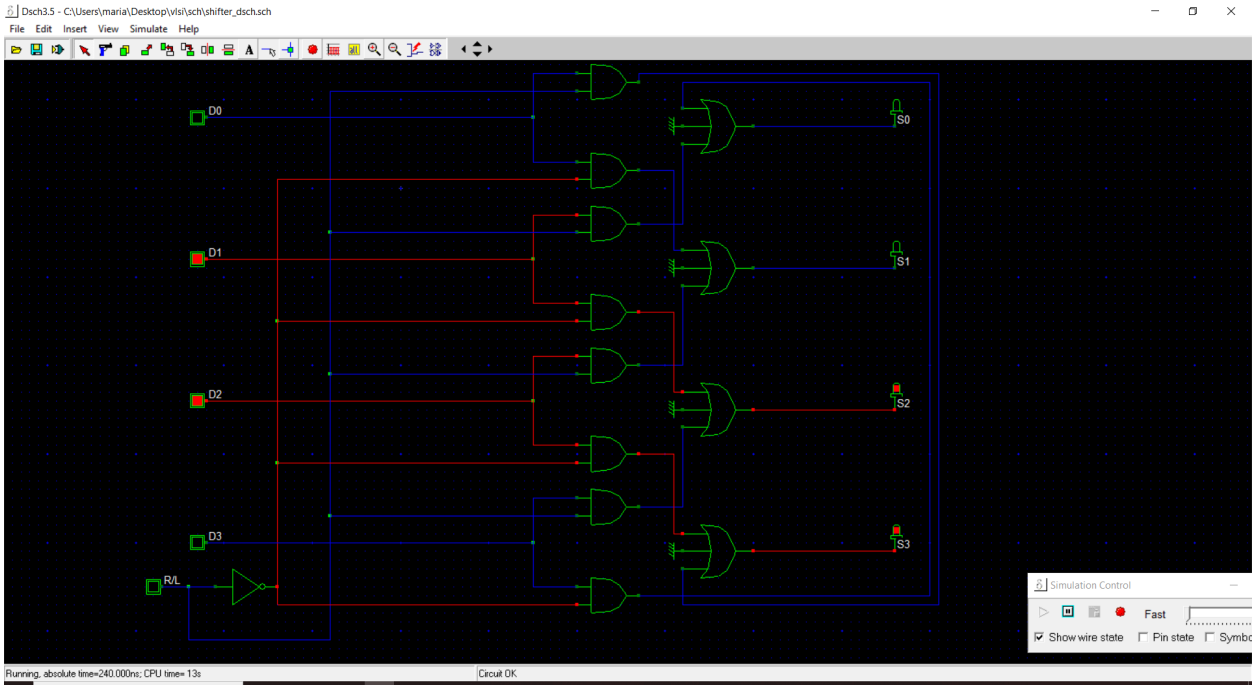


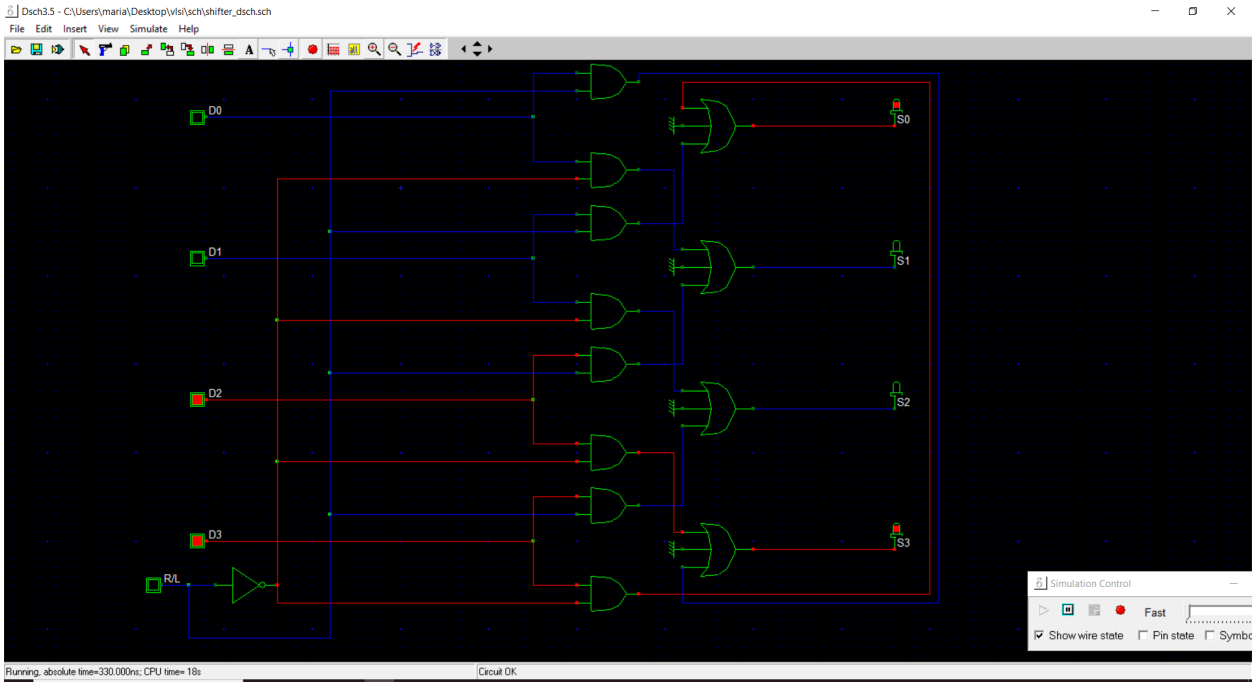
c. Simulare DSCH

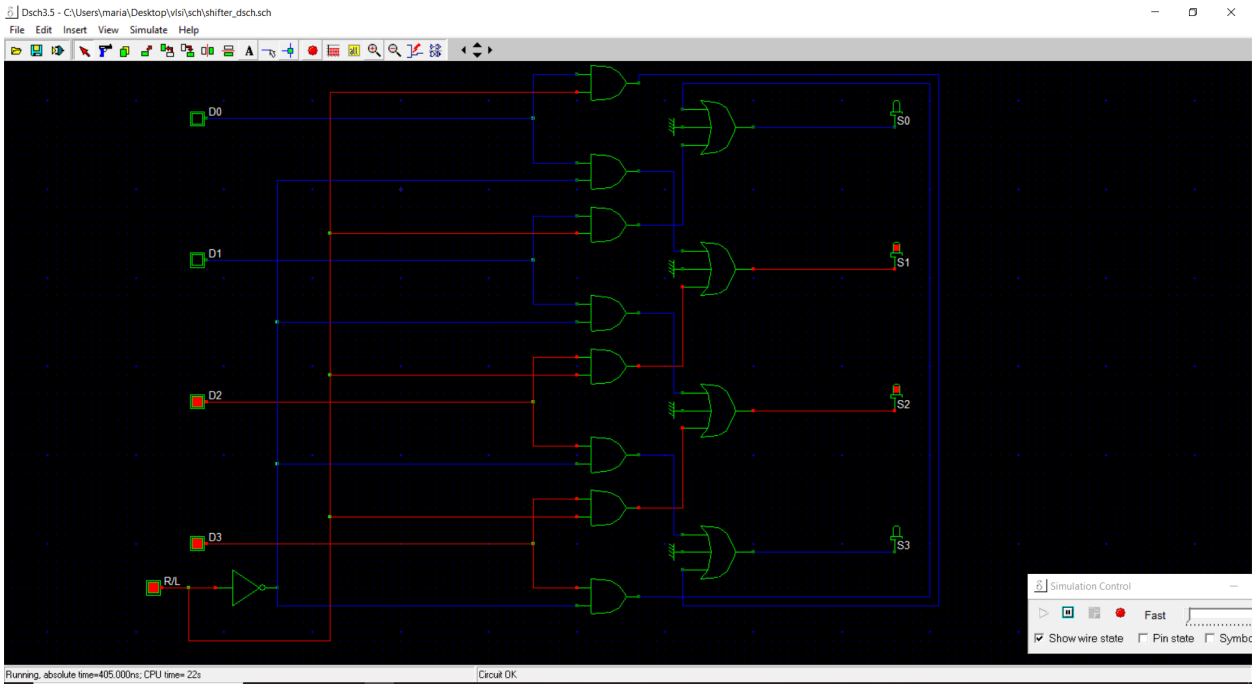


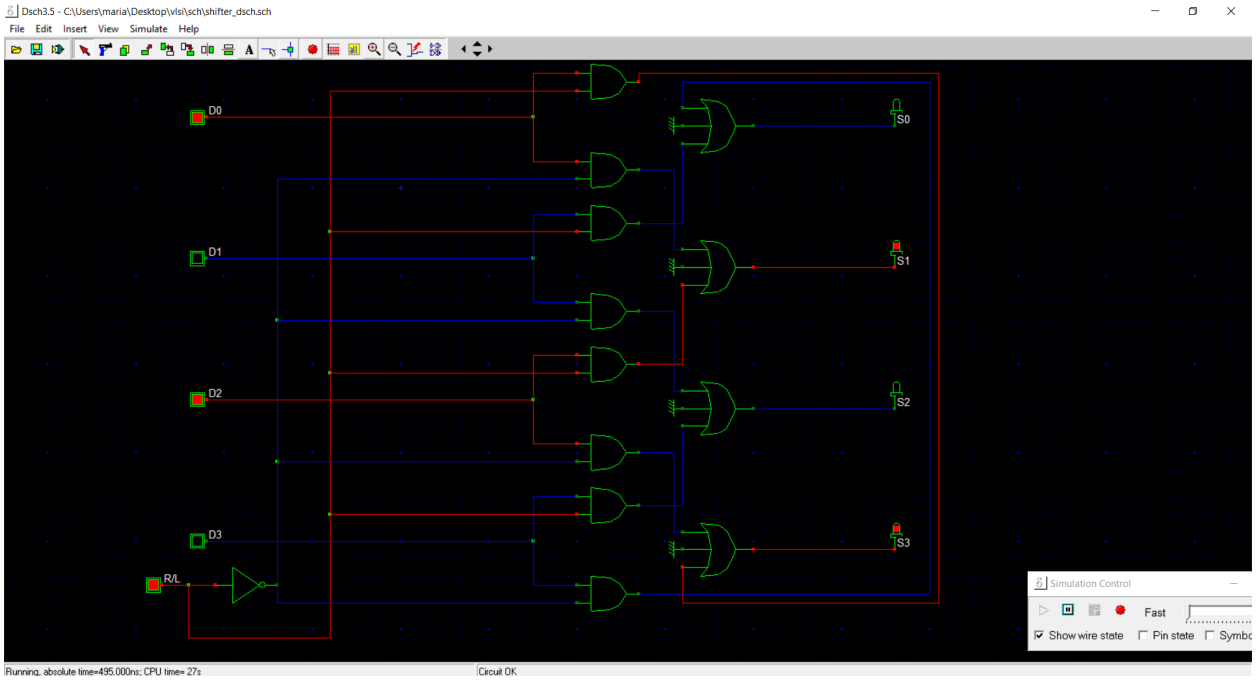



Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\shifter_dsch.sch
File Edit Insert View Simulate Help
D0
D1
D2
D3
R/L
S0
S1
S2
S3
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=165.000ns; CPU time= 9s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\vlsi\sch\shifter_dsch.sch
File Edit Insert View Simulate Help
D0
D1
D2
D3
R/L
S0
S1
S2
S3
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=240.000ns; CPU time= 13s
Circuit OK


Dsch3.5 - C:\Users\maria\Desktop\visi\sch\shifter_dsch.sch
File Edit Insert View Simulate Help
D0
D1
D2
D3
R/L
S0
S1
S2
S3
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=330.000ns; CPU time= 18s
Circuit OK

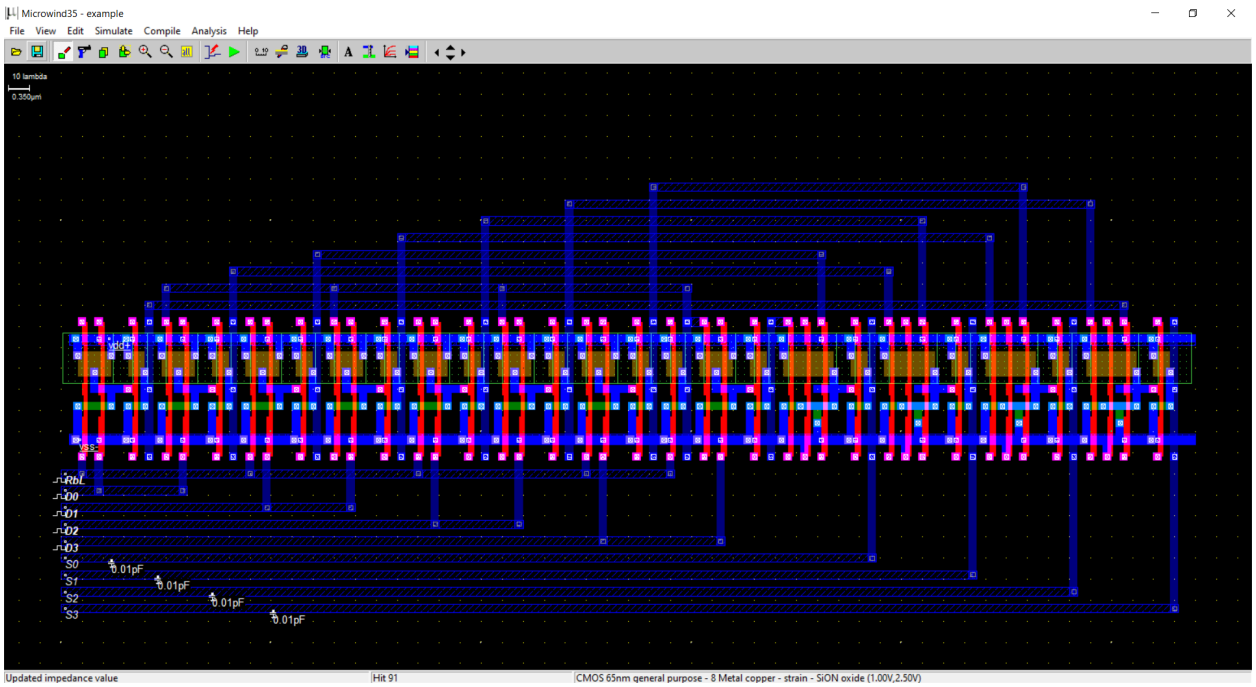

Dsch3.5 - C:\Users\maria\Desktop\visi\sch\shifter_dsch.sch
File Edit Insert View Simulate Help
D0
D1
D2
D3
R/L
S0
S1
S2
S3
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=405.000ns; CPU time= 22s
Circuit OK



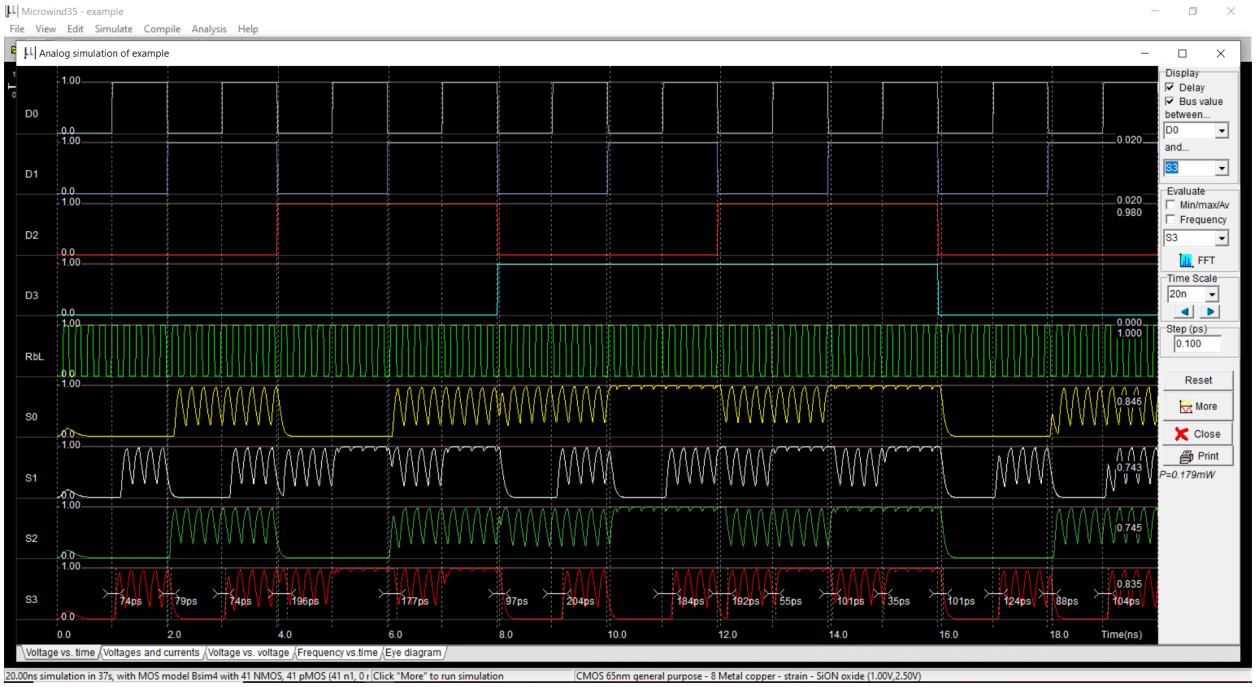
d. Cod Verilog

```
module shifter_dsch( D0,D1,D2,D3,RbL,S2,S1,S0,
 S3);
 input D0,D1,D2,D3,RbL;
 output S2,S1,S0,S3;
 wire w4,w6,w7,w9,w11,w13,w15,w17;
 wire w18,;
 and #(3) and2_1(w4,RbL,D0);
 and #(3) and2_2(w7,w6,D0);
 and #(3) and2_3(w9,RbL,D1);
 and #(3) and2_4(w11,w6,D1);
 and #(3) and2_5(w13,RbL,D2);
 and #(3) and2_6(w15,w6,D2);
 and #(3) and2_7(w17,RbL,D3);
 not #(3) inv_8(w6,RbL);
 and #(3) and2_9(w18,w6,D3);
 or #(4) or3_10(S0,w18,vss,w9);
 or #(4) or3_11(S1,w7,vss,w13);
 or #(4) or3_12(S2,w11,vss,w17);
 or #(4) or3_13(S3,w15,vss,w4);
endmodule

// Simulation parameters in Verilog Format
always
#200 D0=~D0;
#400 D1=~D1;
#800 D2=~D2;
#1600 D3=~D3;
#3200 R/L=~R/L;

// Simulation parameters
// D0 CLK 1 1
// D1 CLK 2 2
// D2 CLK 4 4
// D3 CLK 8 8
// R/L CLK 16 16
```

e. Implementare Microwind

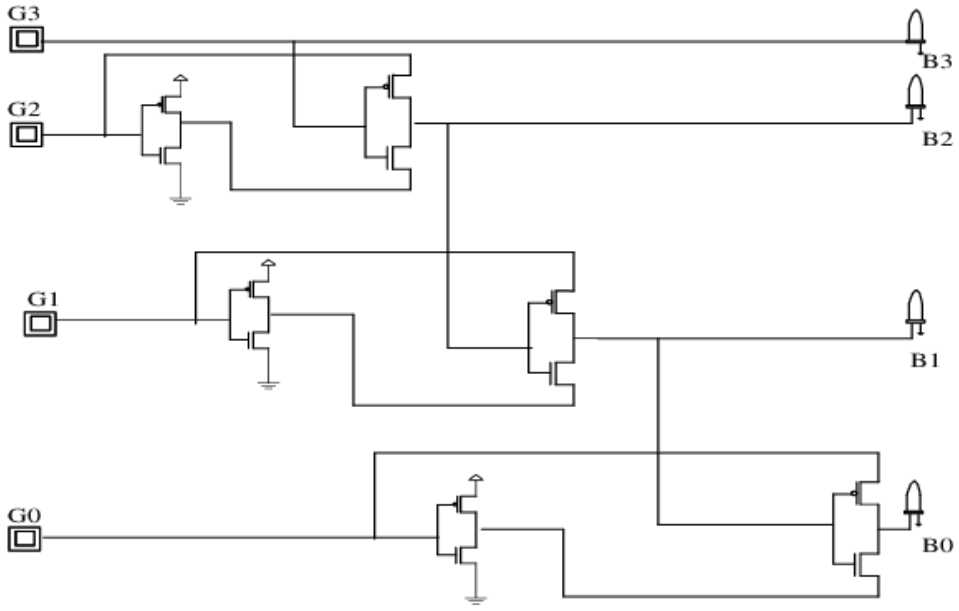


f. Simulare Microwind



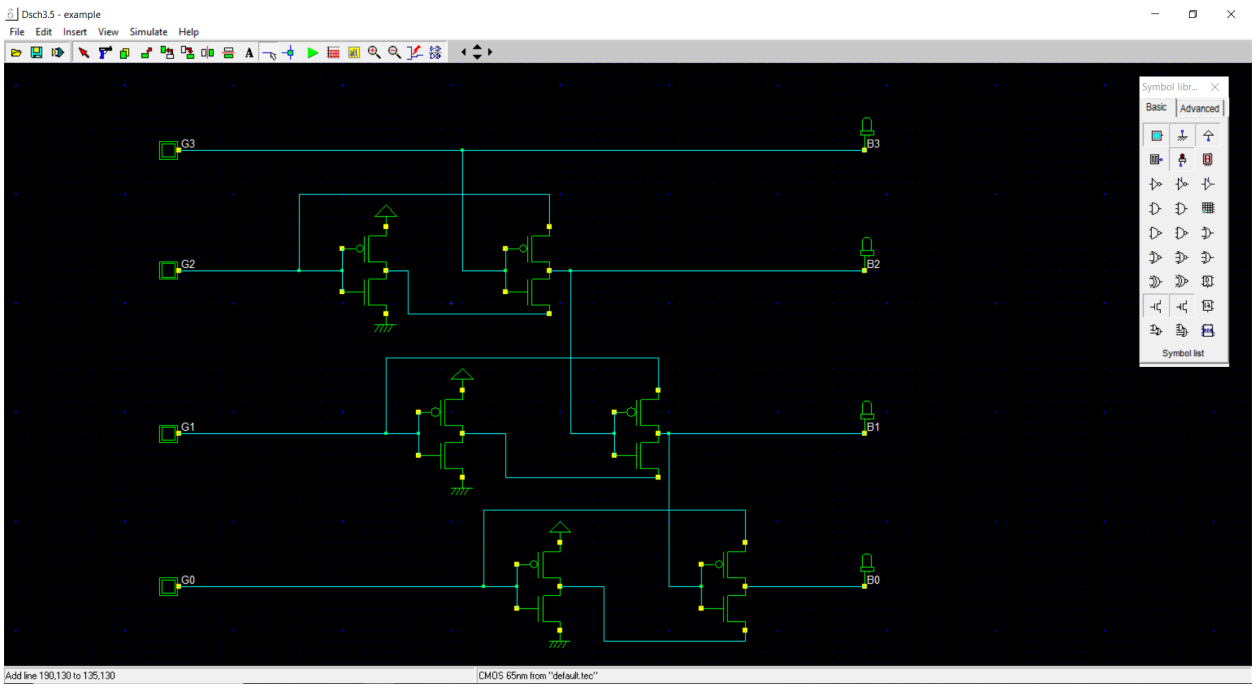
10. Convertor cod Gray 4 biți

a. Schema circuitului



Sursa: https://www.researchgate.net/figure/Waveforms-of-various-MOS-used-in-Binary-Code-to-Gray-Code-Converter-Gray-Code-to-Binary_fig3_304066200

b. Implementare DSCH

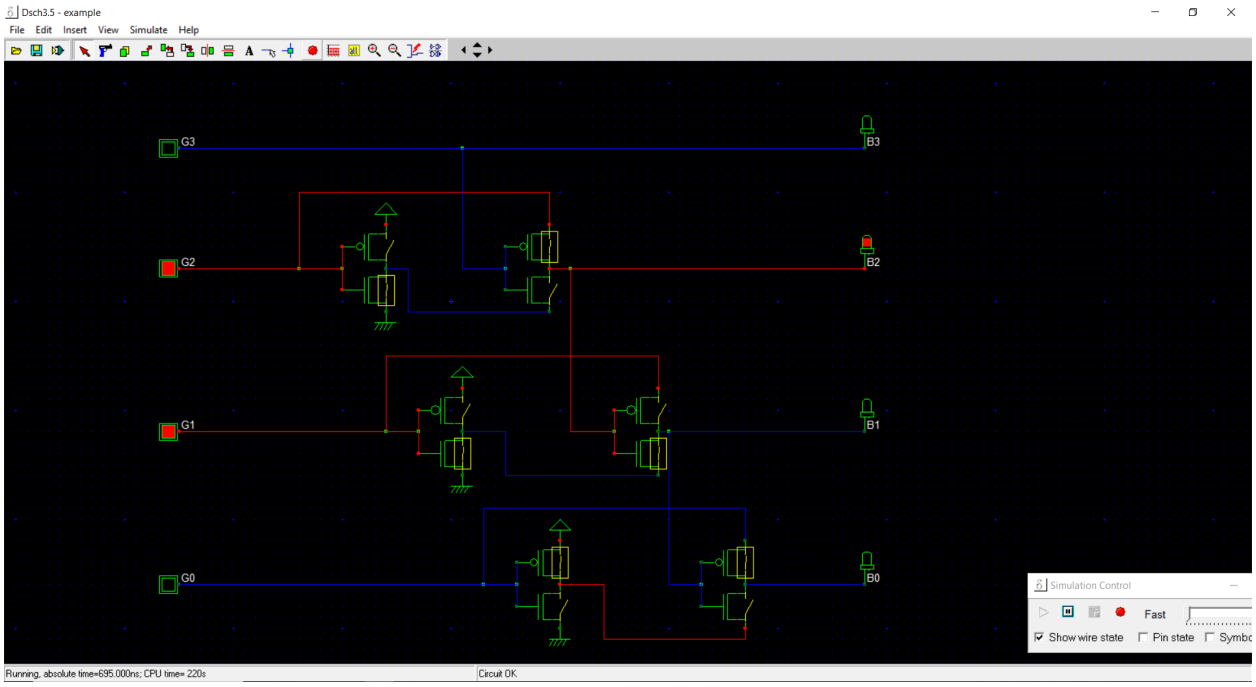


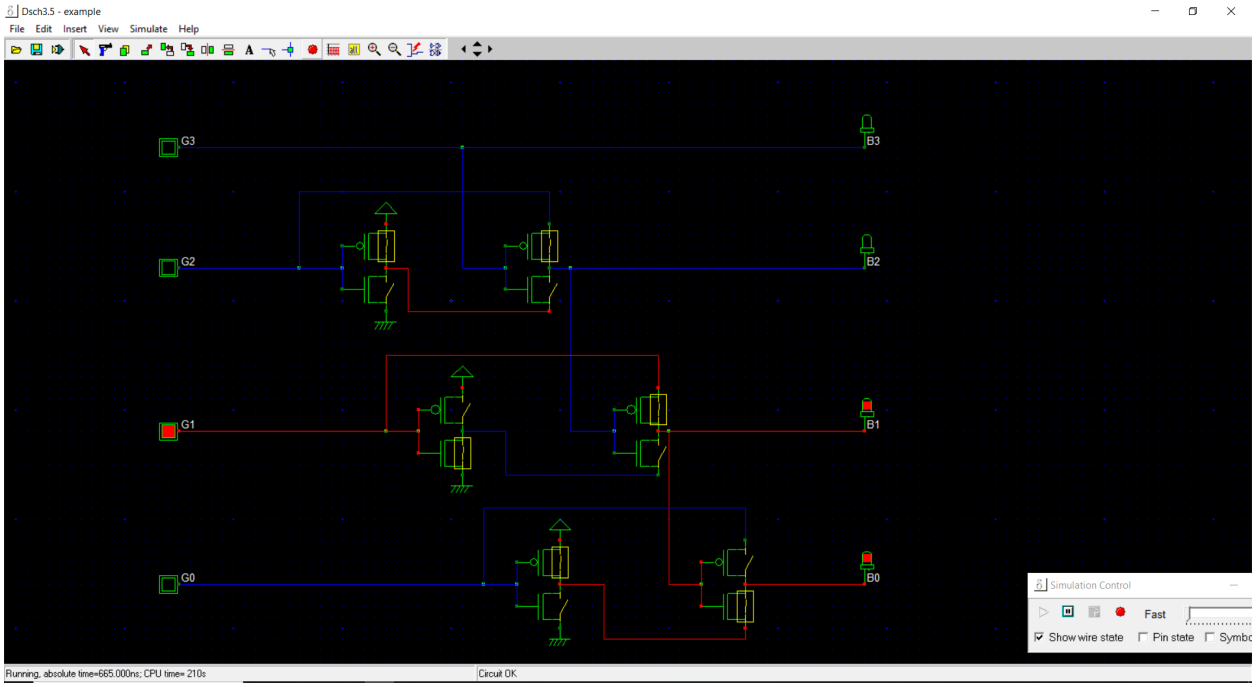
c. Simulare DSCH

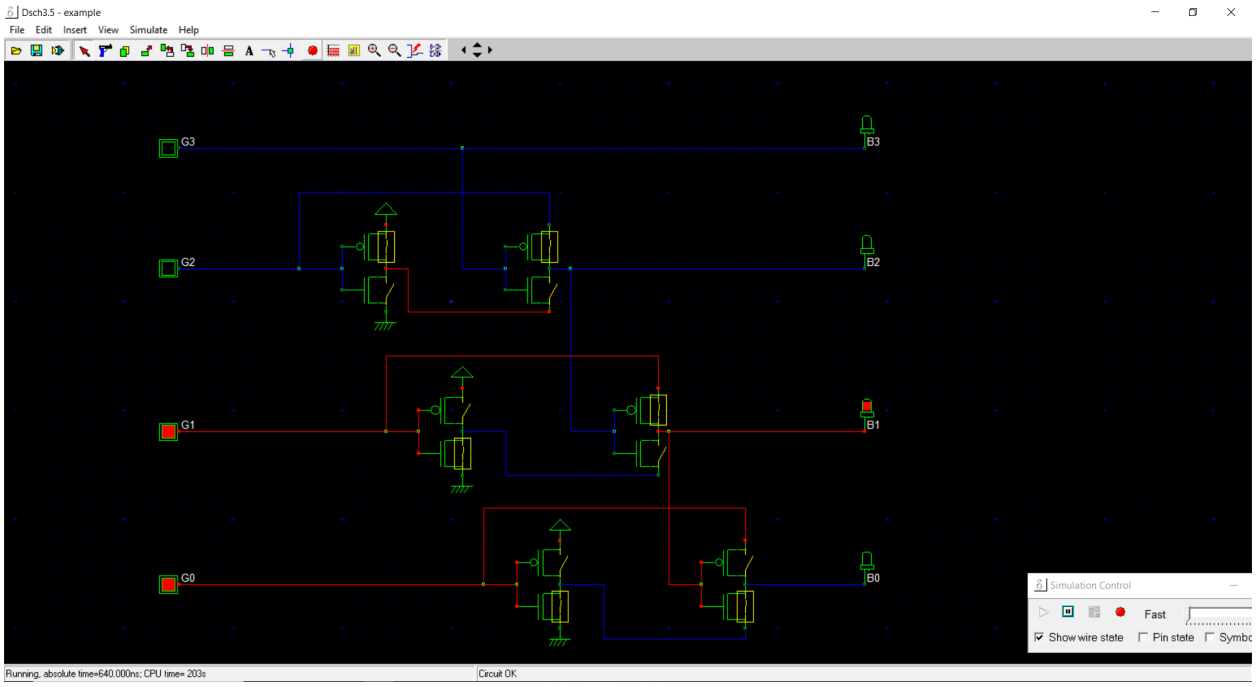
	b[3:0]				g[3:0]			
	0	0	0	0	0	0	0	0
	0	0	0	1	0	0	0	1
	0	0	1	0	0	0	1	1
	0	0	1	1	0	0	1	0
	0	1	0	0	0	1	1	0
	0	1	0	1	0	1	1	1
	0	1	1	0	0	1	0	1
	0	1	1	1	0	1	0	0
	1	0	0	0	1	1	0	0
	1	0	0	1	1	1	0	1
	1	0	1	0	1	1	1	1
	1	0	1	1	1	1	1	0
	1	1	0	0	1	0	1	0
	1	1	0	1	1	0	1	1
	1	1	1	0	1	0	0	1
	1	1	1	1	1	0	0	0

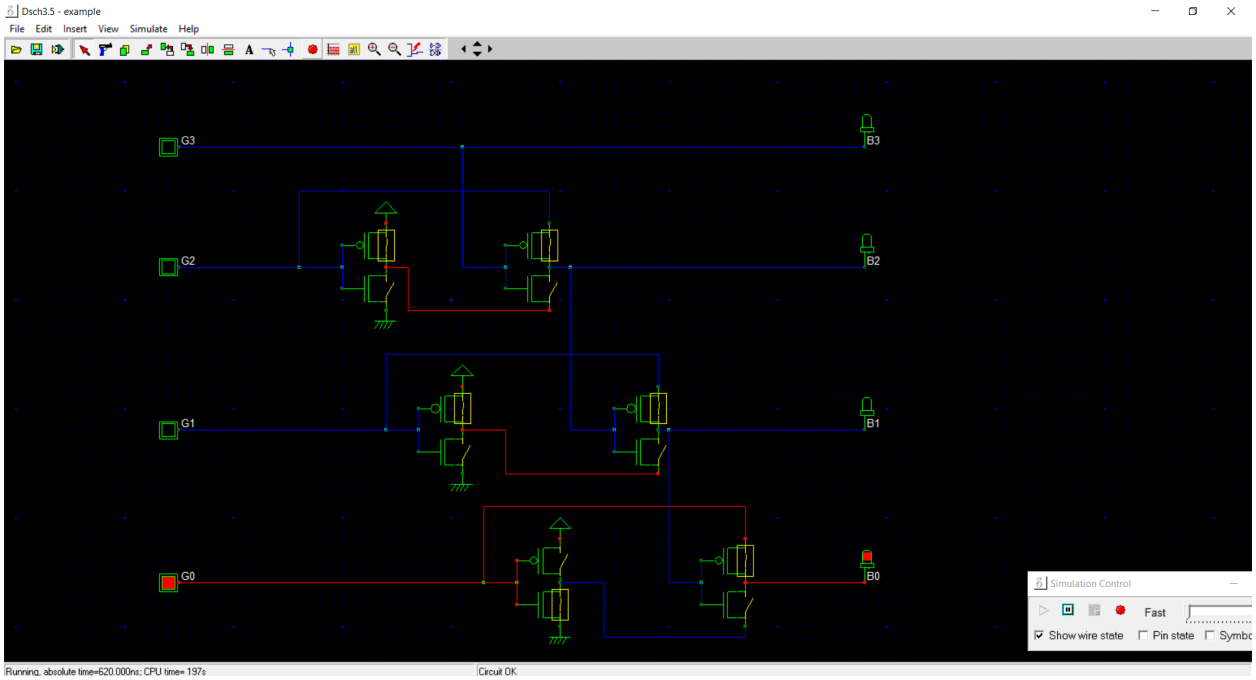
Binary Code →

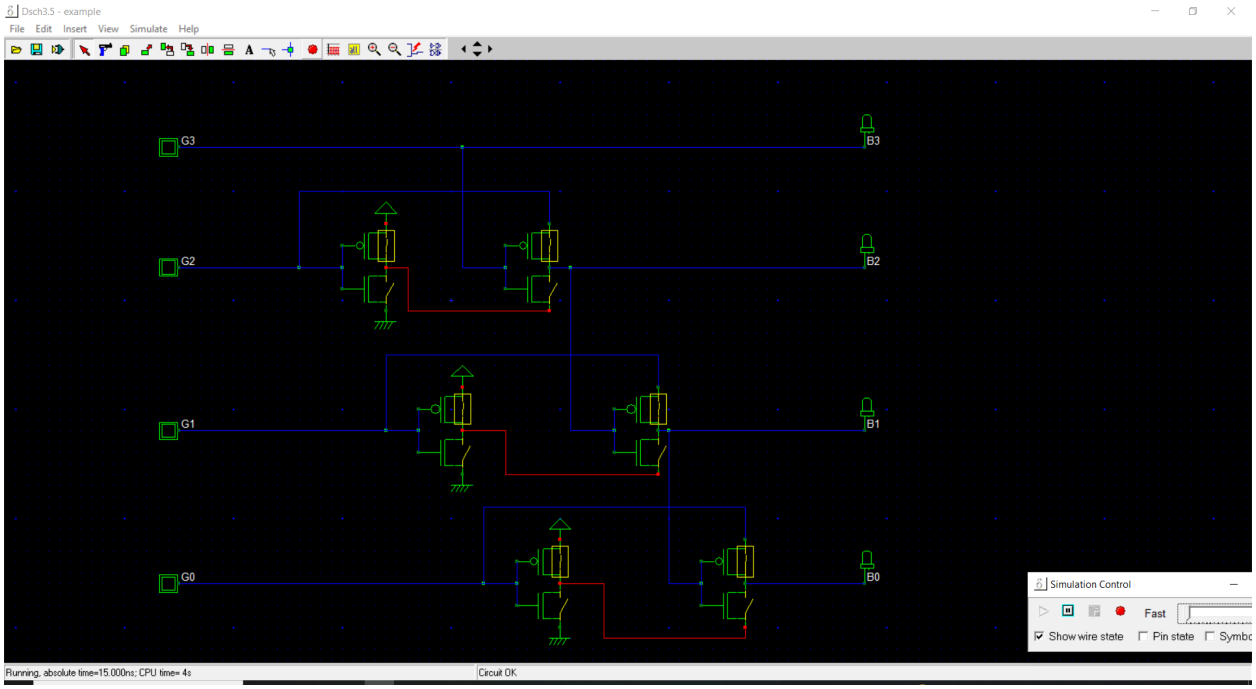
← Gray Code




Dsch3.5 - example
File Edit Insert View Simulate Help
G3
G2
G1
G0
B3
B2
B1
B0
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=665.000ns; CPU time= 210s
Circuit OK


Dsch3.5 - example
File Edit Insert View Simulate Help
G3
G2
G1
G0
B3
B2
B1
B0
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=640.000ns; CPU time= 203s
Circuit OK

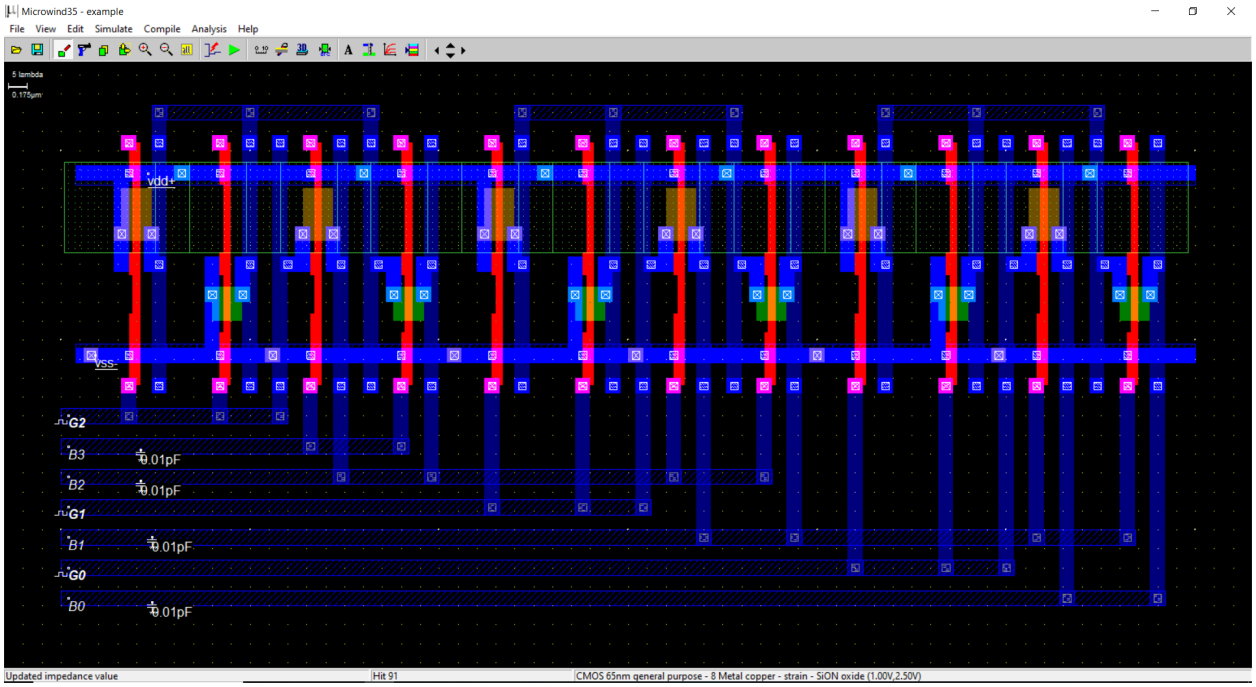

Dsch3.5 - example
File Edit Insert View Simulate Help
G3
G2
G1
G0
B3
B2
B1
B0
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=620.000ns; CPU time= 197s
Circuit OK


Dsch3.5 - example
File Edit Insert View Simulate Help
G3
G2
G1
G0
B3
B2
B1
B0
Simulation Control
Fast
Show wire state
Pin state
Running, absolute time=15.000ns; CPU time= 4s
Circuit OK

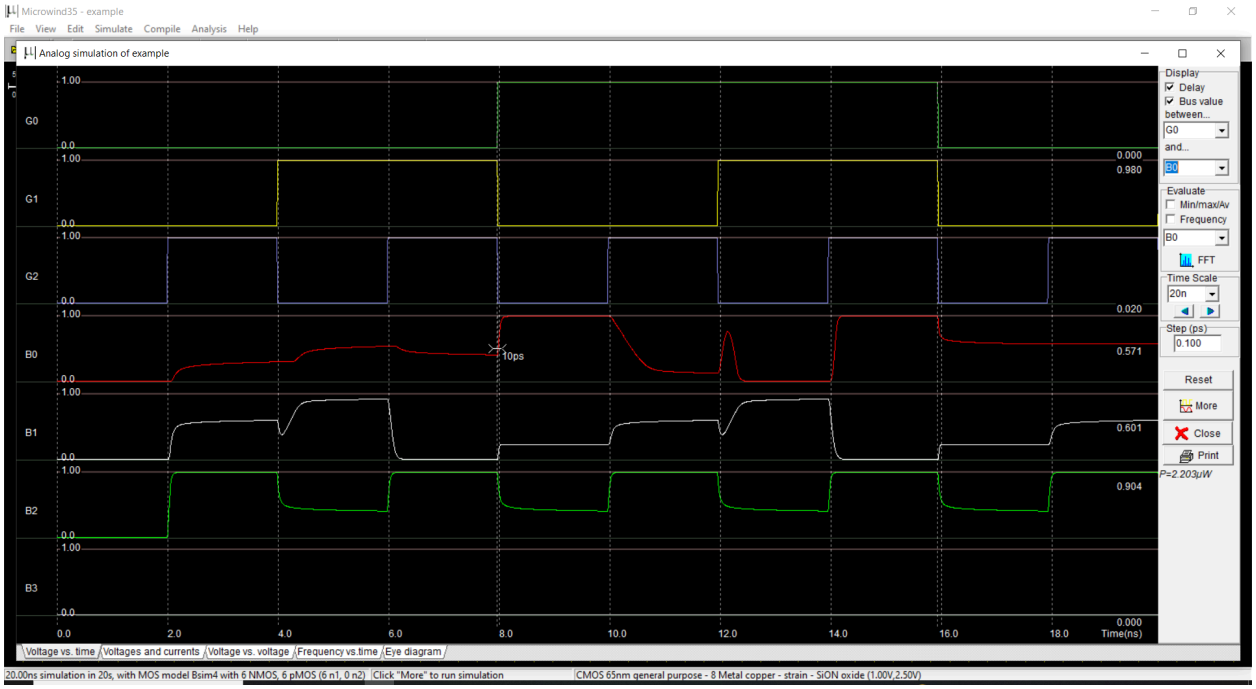
d. Cod Verilog

```
module gray_dsch( G3,G2,G1,G0,B3,B2,B1,B0);
 input G3,G2,G1,G0;
 output B3,B2,B1,B0;
 wire w3,w7,w10,;
 pmos #(2) pmos_1(w3,vdd,G2); // 0.5u 0.07u
 nmos #(2) nmos_2(w3,vss,G2); // 0.3u 0.07u
 pmos #(3) pmos_3(B2,G2,B3); // 0.5u 0.07u
 nmos #(3) nmos_4(B2,w3,B3); // 0.3u 0.07u
 pmos #(2) pmos_5(w7,vdd,G1); // 0.5u 0.07u
 nmos #(2) nmos_6(w7,vss,G1); // 0.3u 0.07u
 pmos #(3) pmos_7(B1,G1,B2); // 0.5u 0.07u
 nmos #(3) nmos_8(B1,w7,B2); // 0.3u 0.07u
 pmos #(2) pmos_9(w10,vdd,G0); // 0.5u 0.07u
 nmos #(2) nmos_10(w10,vss,G0); // 0.3u 0.07u
 pmos #(2) pmos_11(B0,G0,B1); // 0.5u 0.07u
 nmos #(2) nmos_12(B0,w10,B1); // 0.3u 0.07u
endmodule

// Simulation parameters in Verilog Format
always
#200 G3=~G3;
#400 G2=~G2;
#800 G1=~G1;
#1600 G0=~G0;

// Simulation parameters
// G3 CLK 1 1
// G2 CLK 2 2
// G1 CLK 4 4
// G0 CLK 8 8
```

e. Implementare Microwind

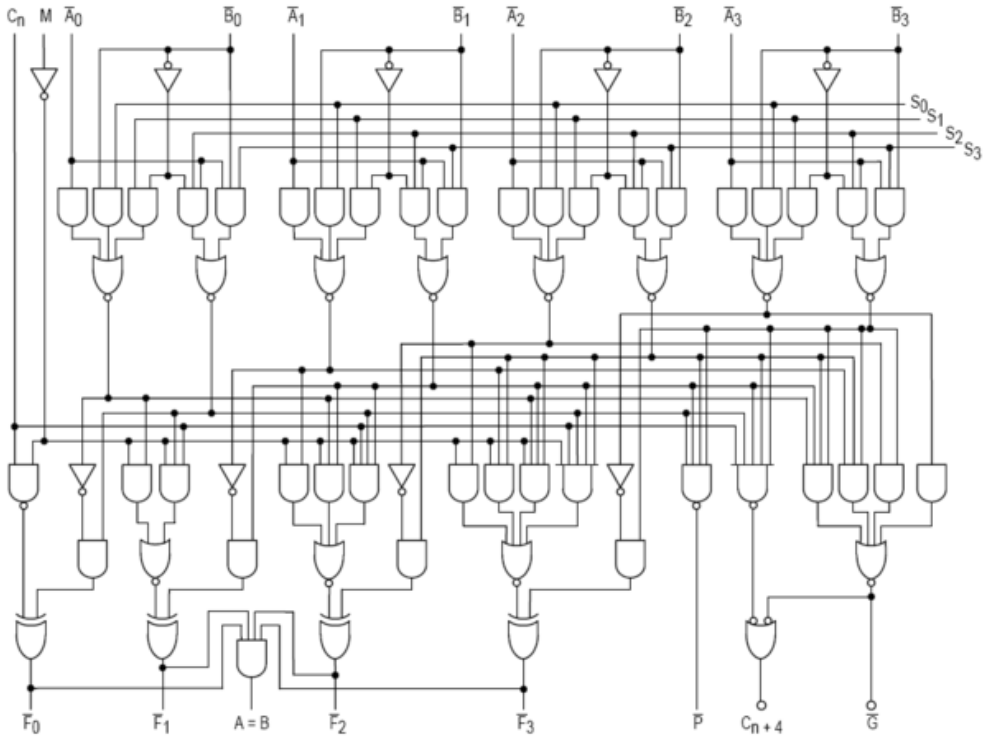


f. Simulare Microwind



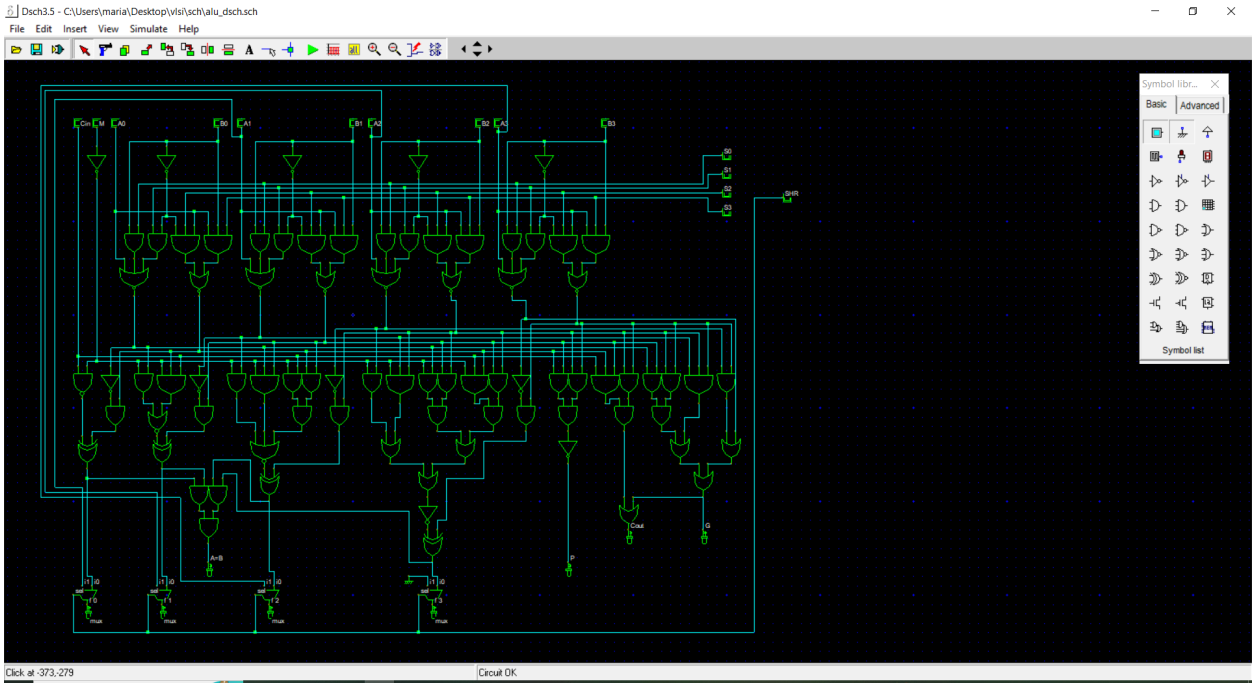
11. ALU pe 4 biți (not, +, -, nor, shr, shl)

a. Schema circuitului



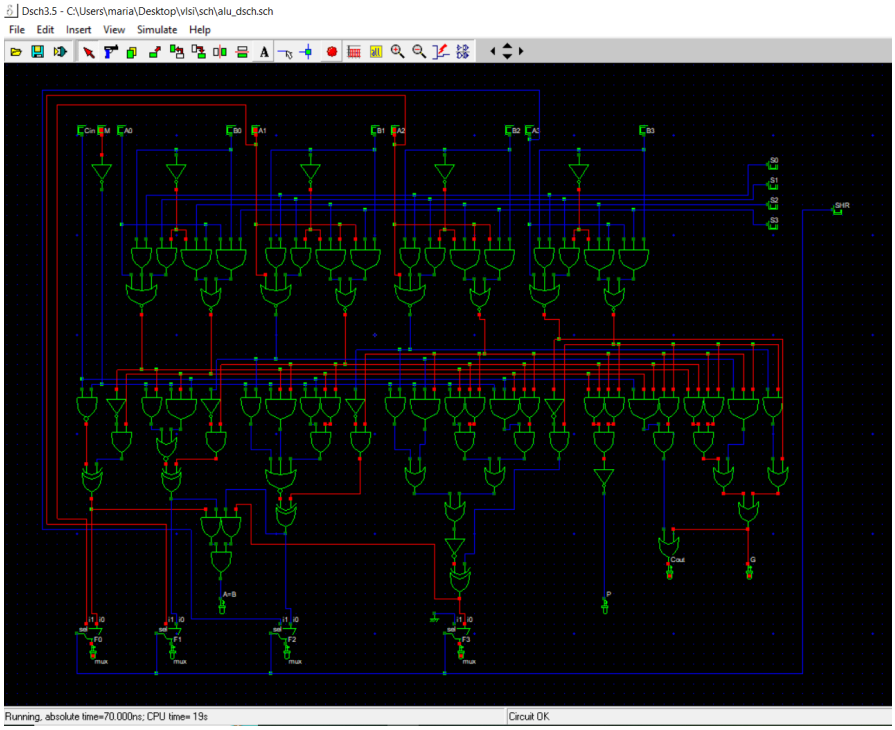
Sursa: https://en.wikipedia.org/wiki/74181

b. Implementare DSCH



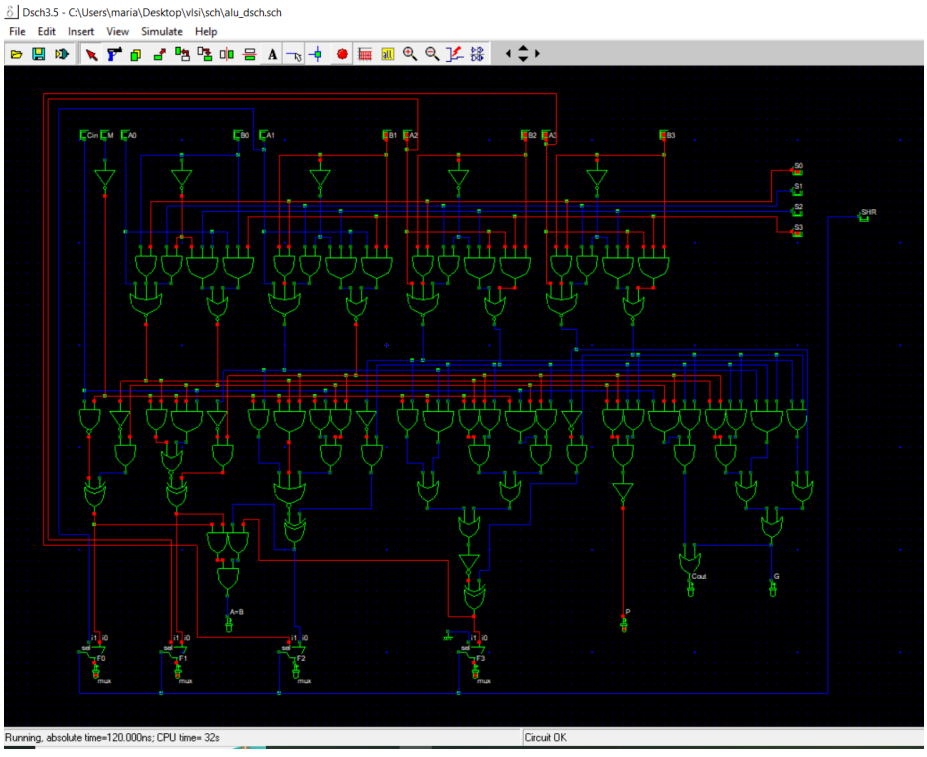
c. Simulare DSCH

NOT: Cin = 0, M = 1, S0 = 0, S1 = 0, S2 = 0, S3 = 0



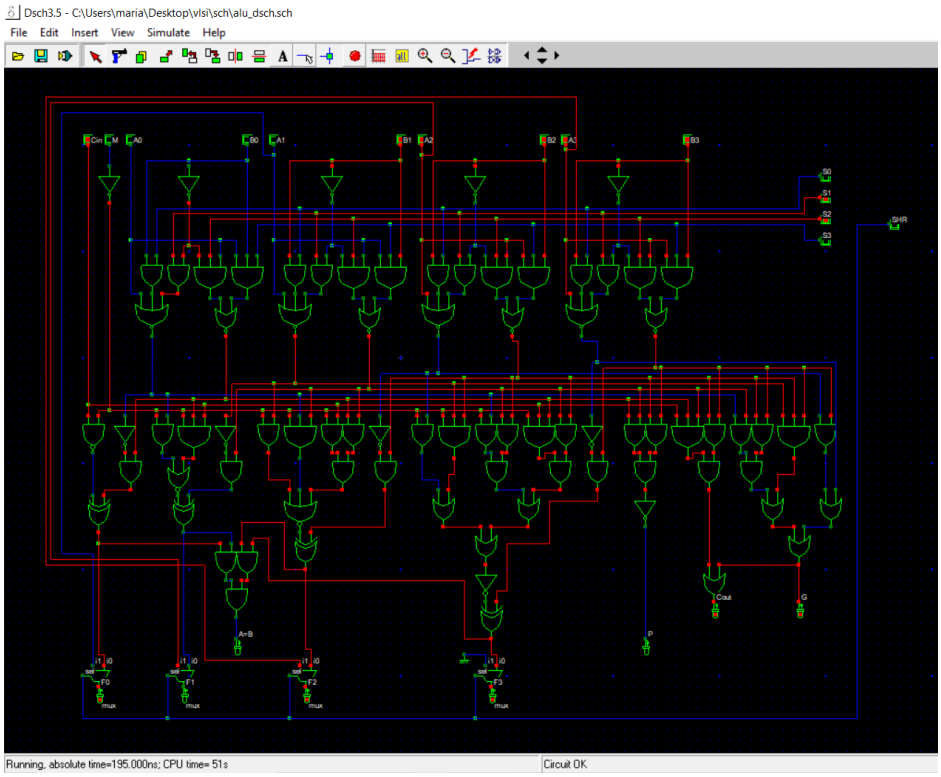
+: Cin = 0, M = 0, S0 = 1, S1 = 0, S2 = 0, S3 = 1

Obs.: A, B și F sunt negate.

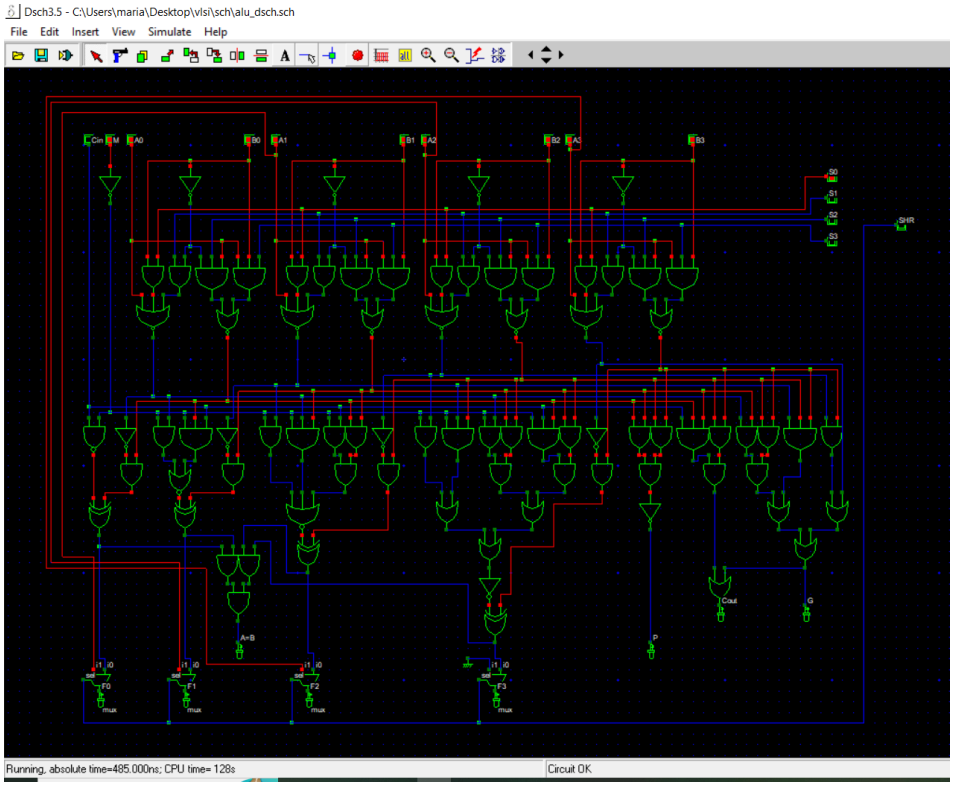


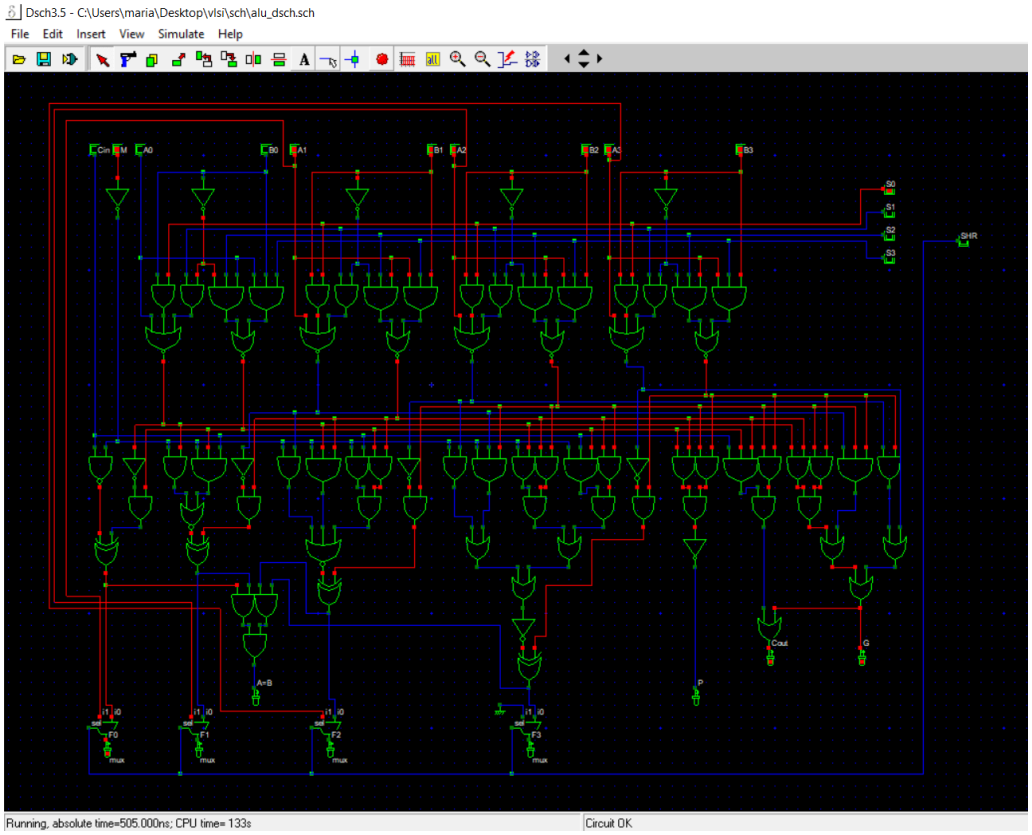
-: Cin = 1, M = 0, S0 = 0, S1 = 1, S2 = 1, S3 = 0

Obs.: A, B și F sunt negate.

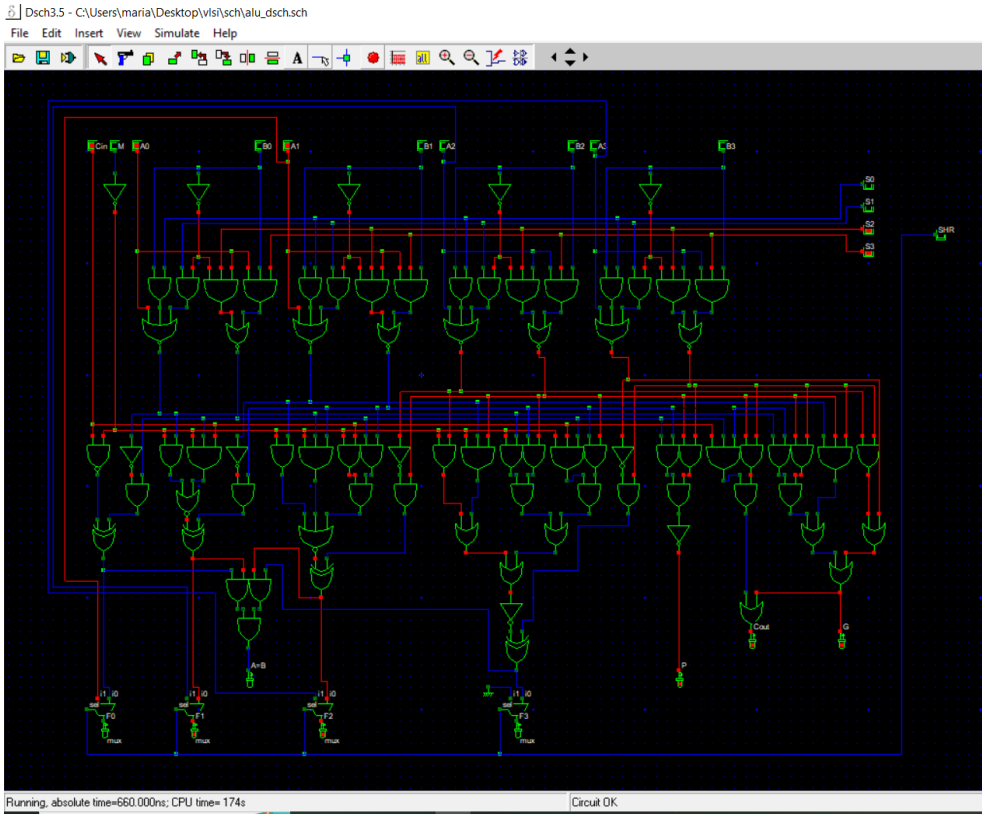


NOR: Cin = 0, M = 1, S0 = 1, S1 = 0, S2 = 0, S3 = 0

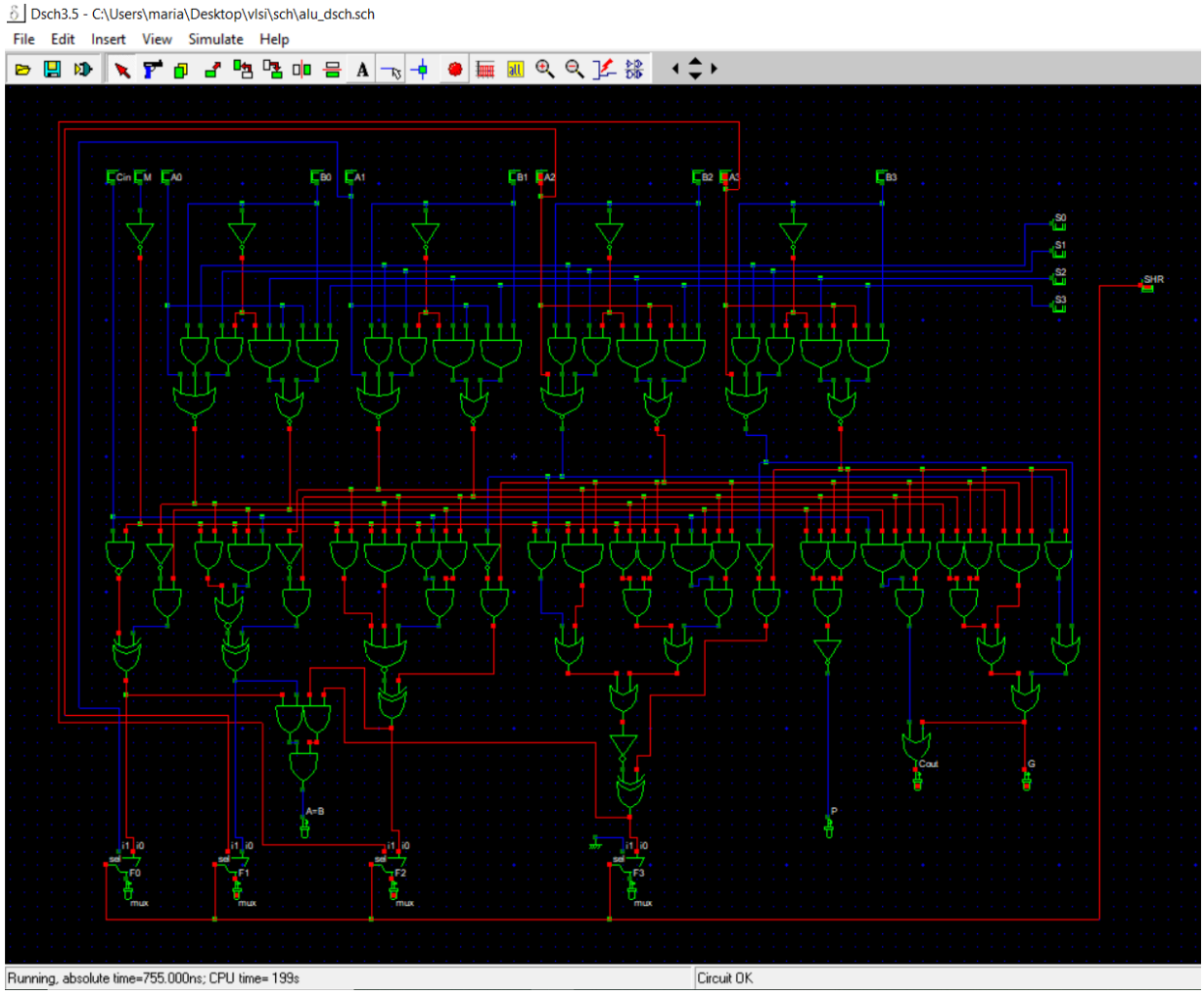




SHL: Cin = 1, M = 0, S0 = 0, S1 = 0, S2 = 1, S3 = 1



SHR: buton SHR

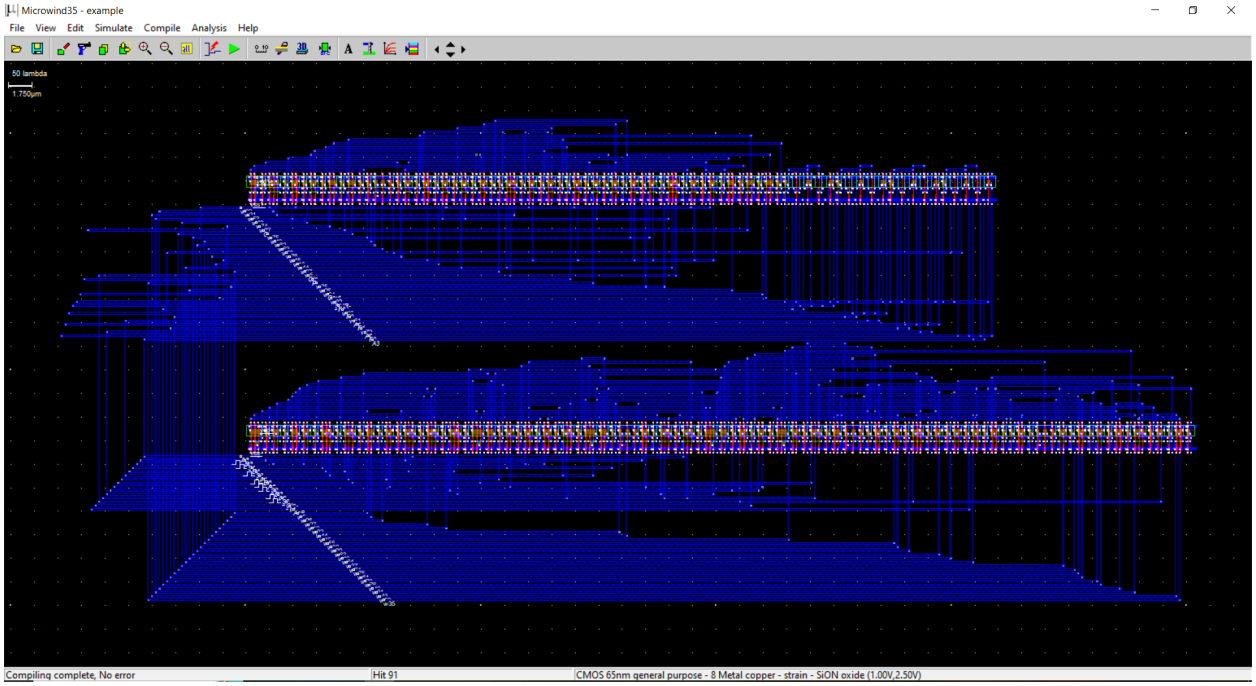


d. Cod Verilog

```
module alu_dsch(
Cin,M,B2,S3,A0,B0,A1,B1,
 A2,A3,B3,S0,S1,S2,SHR,F3,
 F0,F1,G,F2,AeB,P,Cout);
 input Cin,M,B2,S3,A0,B0,A1,B1;
 input A2,A3,B3,S0,S1,S2,SHR;
 output F3,F0,F1,G,F2,AeB,P,Cout;
 wire w2,w3,w5,w10,w12,w13,w14,w15;
 wire
w16,w17,w18,w22,w25,w26,w27,w29;
 wire
w30,w31,w32,w34,w35,w36,w37,w38;
 wire
w39,w40,w42,w43,w44,w45,w46,w47;
 wire
w48,w49,w50,w51,w52,w53,w54,w55;
 wire
w56,w57,w58,w59,w60,w61,w62,w63;
 wire
w64,w65,w66,w68,w69,w70,w71,w72;
 wire
w73,w74,w75,w76,w77,w78,w79,w80;
 wire
w81,w82,w83,w85,w87,w88,w89,w90;
 wire w91,w95,w96,w97,w98,w99,;
 nor #(3) nor3_1(w5,w2,w3,A3);
 not #(2) inv_2(w10,B3);
 and #(3) and2_3(w13,S1,w12);
 nor #(6) nor2_4(w16,w14,w15);
 and #(3) and2_5(w18,S1,w17);
 and #(3) and3_6(w22,B1,S3,A1);
 and #(3) and3_7(w25,S3,B0,A0);
 or #(3) or2_8(w27,w5,w26);
 and #(3) and2_9(w29,B0,S0);
 and #(3) and3_10(w14,B3,S3,A3);
 and #(3) and3_11(w32,w16,w30,w31);
 and #(3) and3_12(w34,B2,S3,A2);
 nor #(4) nor3_13(w35,w18,w29,A0);
 nor #(5) nor2_14(w37,w25,w36);
 not #(2) inv_15(w17,B0);
 and #(3) and2_16(w38,B1,S0);
 and #(3) and2_17(w2,S1,w10);
 and #(3) and2_18(w40,S1,w39);
 and #(3) and3_19(w42,A1,S2,w12);
 nor #(4) nor3_20(w31,w13,w38,A1);
 nor #(6) nor2_21(w43,w22,w42);
 not #(2) inv_22(w12,B1);
 xor #(4) xor2_23(w46,w44,w45);
 and #(3) and2_24(w3,B3,S0);
 and #(3) and3_25(w47,A2,S2,w39);
 and #(3) and3_26(w15,A3,S2,w10);
 nor #(4) nor3_27(w49,w40,w48,A2);
 nor #(3) nor2_28(w52,w50,w51);
 not #(2) inv_29(w39,B2);
 nand #(2) nand2_30(w54,Cin,w53);
 not #(5) inv_31(w53,M);
 and #(3) and2_32(w48,B2,S0);
 nor #(2) nor3_33(w58,w55,w56,w57);
 and #(3) and2_34(w60,w59,w37);
 and #(3) and3_35(w36,A0,S2,w17);
 nor #(7) nor2_36(w30,w34,w47);
 xor #(4) xor2_37(w61,w60,w54);
 not #(1) inv_38(w59,w35);
 and #(3) and2_39(w51,w53,w35);
 and #(3) and2_40(w63,w62,w30);
 and #(3) and3_41(w50,Cin,w37,w53);
 and #(3) and2_42(w64,w53,w49);
 and #(3) and2_43(AeB,w65,w66);
 and #(3) and2_44(w68,w37,w43);
 and #(3) and2_45(w55,w69,w68);
 not #(1) inv_46(w70,w31);
 and #(3) and2_47(w71,w70,w43);
 xor #(4) xor2_48(w72,w71,w52);
 and #(3) and2_49(w75,w73,w74);
```

```
and #(3) and2_50(w65,w61,w72);
and #(3) and2_51(w66,w76,w46);
and #(3) and2_52(w57,w53,w31);
and #(3) and3_53(w56,w43,w35,w53);
and #(3) and2_54(w69,w53,Cin);
and #(3) and2_55(w73,w53,w35);
or #(3) or2_56(w79,w77,w78);
and #(3) and3_57(w80,w30,w31,w53);
and #(3) and2_58(w74,w43,w30);
and #(3) and2_59(w83,w81,w82);
or #(4) or2_60(G,w27,w85);
or #(3) or2_61(w77,w87,w75);
not #(1) inv_62(w62,w49);
xor #(4) xor2_63(w76,w63,w58);
not #(1) inv_64(w45,w79);
and #(3) and2_65(w90,w88,w89);
and #(3) and2_66(w44,w91,w16);
and #(3) and2_67(w95,w43,w30);
and #(3) and2_68(w81,w37,w43);
and #(4) and2_69(w98,w96,w97);
not #(1) inv_70(w91,w5);
and #(3) and3_71(w99,w37,Cin,w53);
and #(3) and2_72(w82,w30,w16);
and #(3) and2_73(w87,w99,w95);
not #(1) inv_74(P,w83);
and #(3) and2_75(w89,w30,w16);
and #(4) and2_76(w96,w35,w43);
and #(4) and2_77(w98,w96,w97);
and #(4) and2_78(w26,w49,w16);
and #(4) and2_79(w26,w49,w16);
and #(3) and3_80(w88,w43,w37,Cin);
and #(4) and2_81(w97,w30,w16);
and #(4) and2_82(w97,w30,w16);
or #(3) or2_83(w78,w80,w64);
or #(3) or2_84(Cout,G,w90);
or #(3) or2_85(w85,w32,w98);
mux #(1) mux_86(F3,w46,vss,SHR);
mux #(1) mux_87(F0,w61,A1,SHR);
 mux #(1) mux_88(F1,w72,A2,SHR);
 mux #(1) mux_89(F2,w76,A3,SHR);
endmodule

// Simulation parameters in Verilog Format
always
#200 Cin=~Cin;
#400 M=~M;
#800 B2=~B2;
#1600 S3=~S3;
#3200 A0=~A0;
#6400 B0=~B0;
#12800 A1=~A1;
#25600 B1=~B1;
#51200 A2=~A2;
#102400 A3=~A3;
#102400 B3=~B3;
#102400 S0=~S0;
#102400 S1=~S1;
#102400 S2=~S2;
#102400 SHR=~SHR;

// Simulation parameters
// Cin CLK 1 1
// M CLK 2 2
// B2 CLK 4 4
// S3 CLK 8 8
// A0 CLK 16 16
// B0 CLK 32 32
// A1 CLK 64 64
// B1 CLK 128 128
// A2 CLK 256 256
// A3 CLK 512 512
// B3 CLK 1024 1024
// S0 CLK 2048 2048
// S1 CLK 4096 4096
// S2 CLK 8192 8192
// SHR CLK 16384 16384
```

e. Implementare Microwind



f. Simulare Microwind

(Simulare pentru NOR)