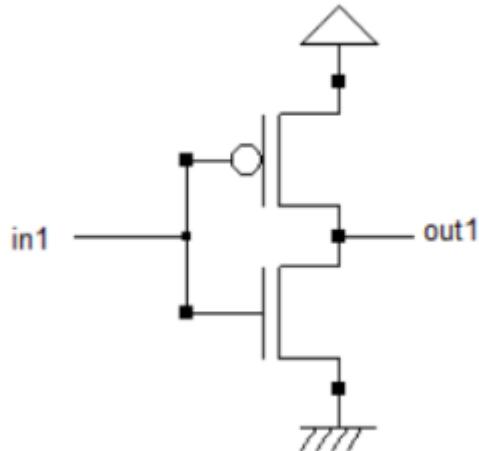


Referat VLSI

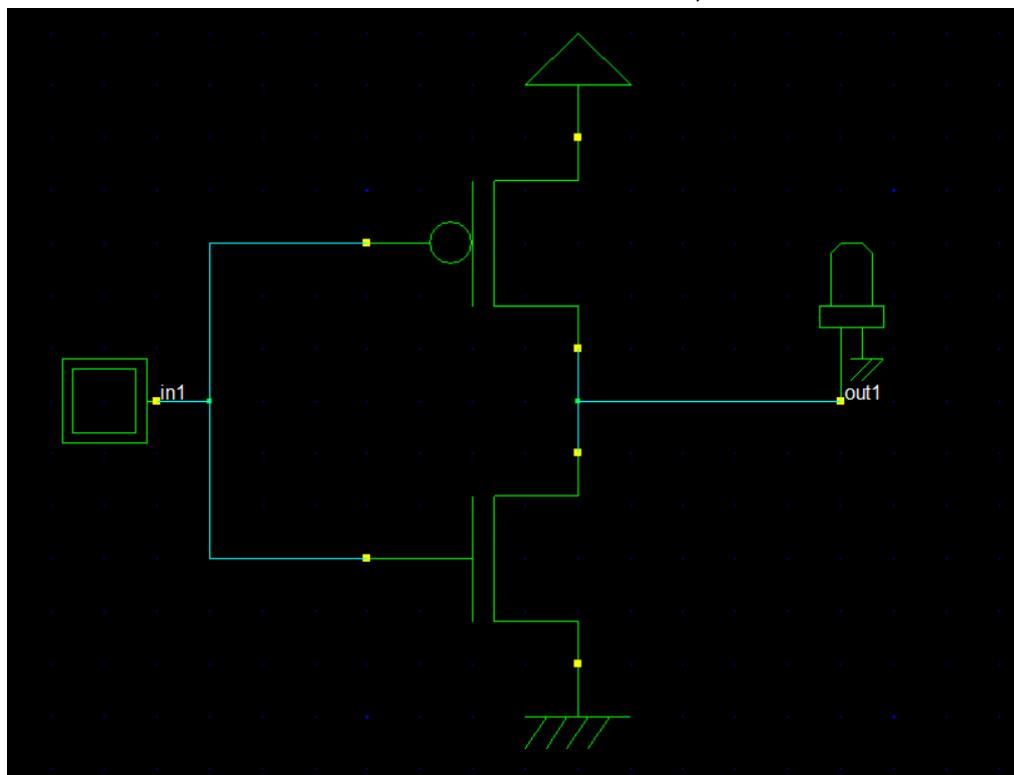
1. Inversor

Inversorul reprezintă unitatea de bază pentru orice circuit digital.

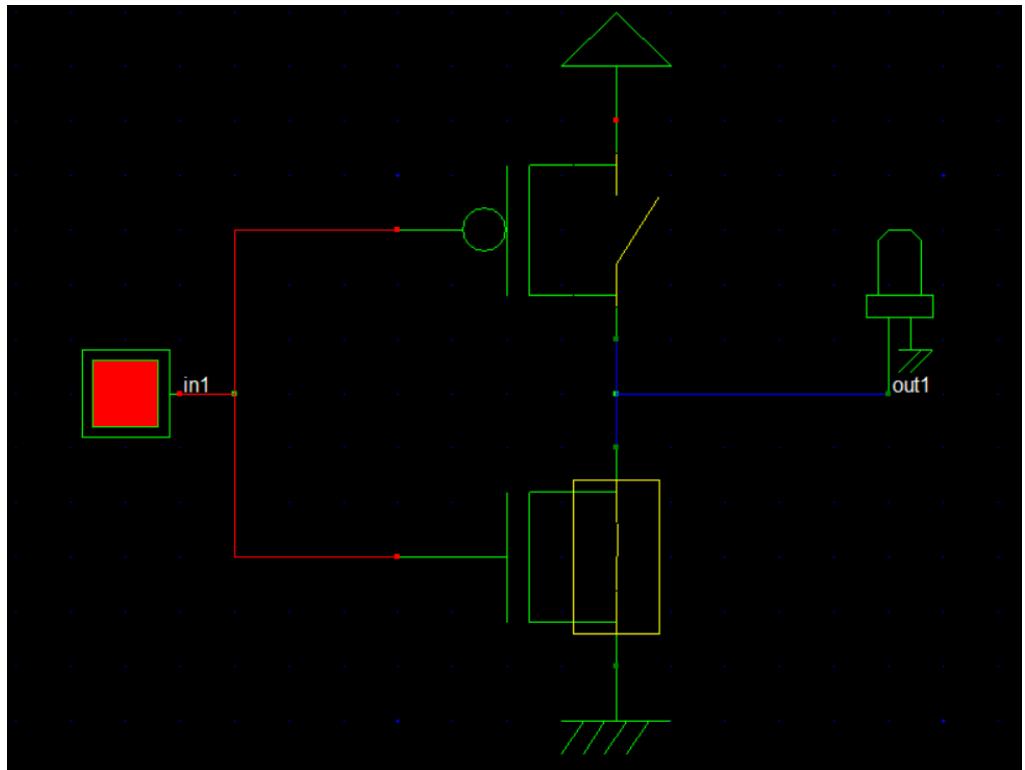
a. Schema circuitului – (Cartea Circuite Integrate la Scara Larga)



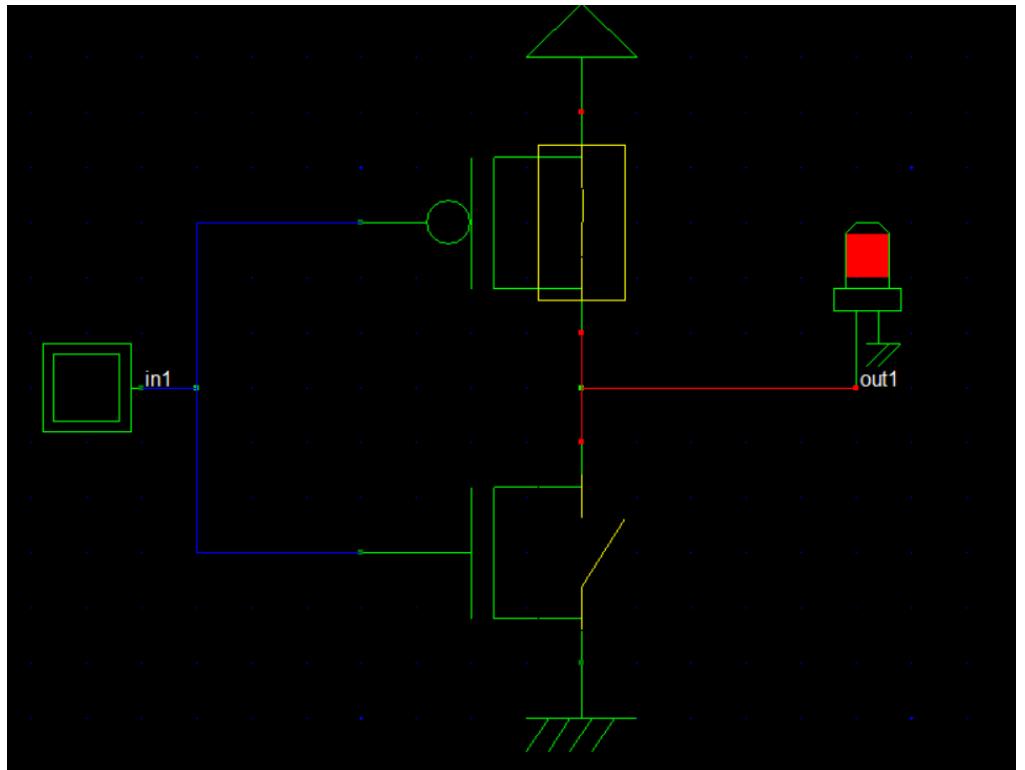
b. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS



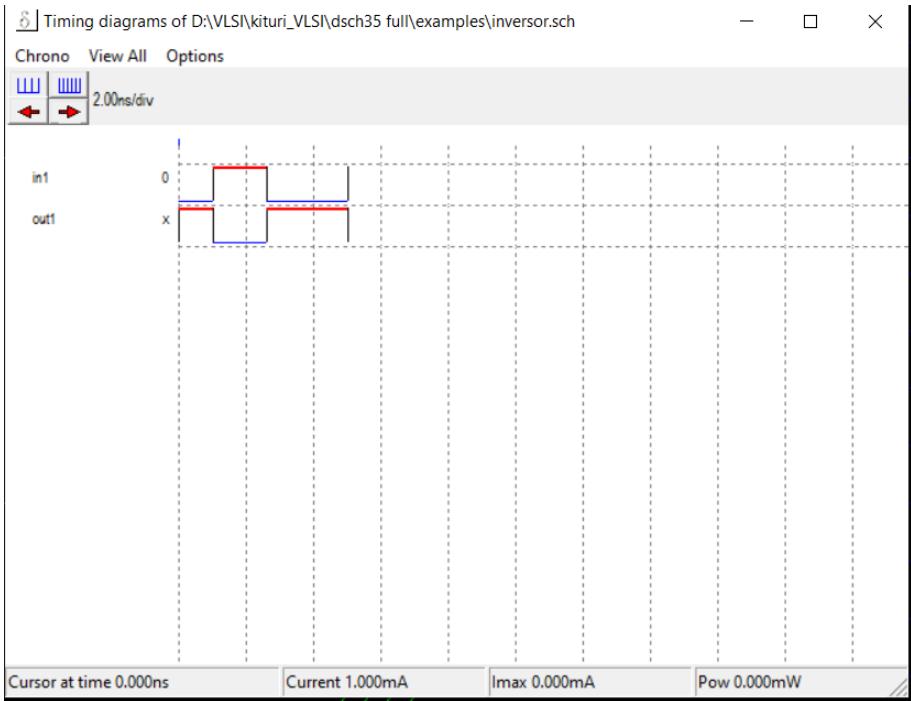
c. Simularea în DSCH



Input 1, Output 0



Input 0, Output 1



Timing diagrams

d. Codul Verilog obținut

The figure shows a Verilog code generation interface. On the left, there is a code editor with tabs for Verilog, Hierarchy, Netlist, and Critical path. The Verilog tab contains the following code:

```

// DSCH 3.5
// 5/10/2023 10:57:42 PM
// D:\VLSI\kituri_VLSI\dsch35 full\examples\invensor.sch

module invensor( 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

```

On the right, there is an "Information" panel with the following details:

- Module name (8 char. max): invensor
- Add gate delay info
- Append simul. infomations
- Add labels as comments

Information summary:

- The Verilog file has 18 lines
- The design includes 6 symbols
- The circuit has 3 nodes

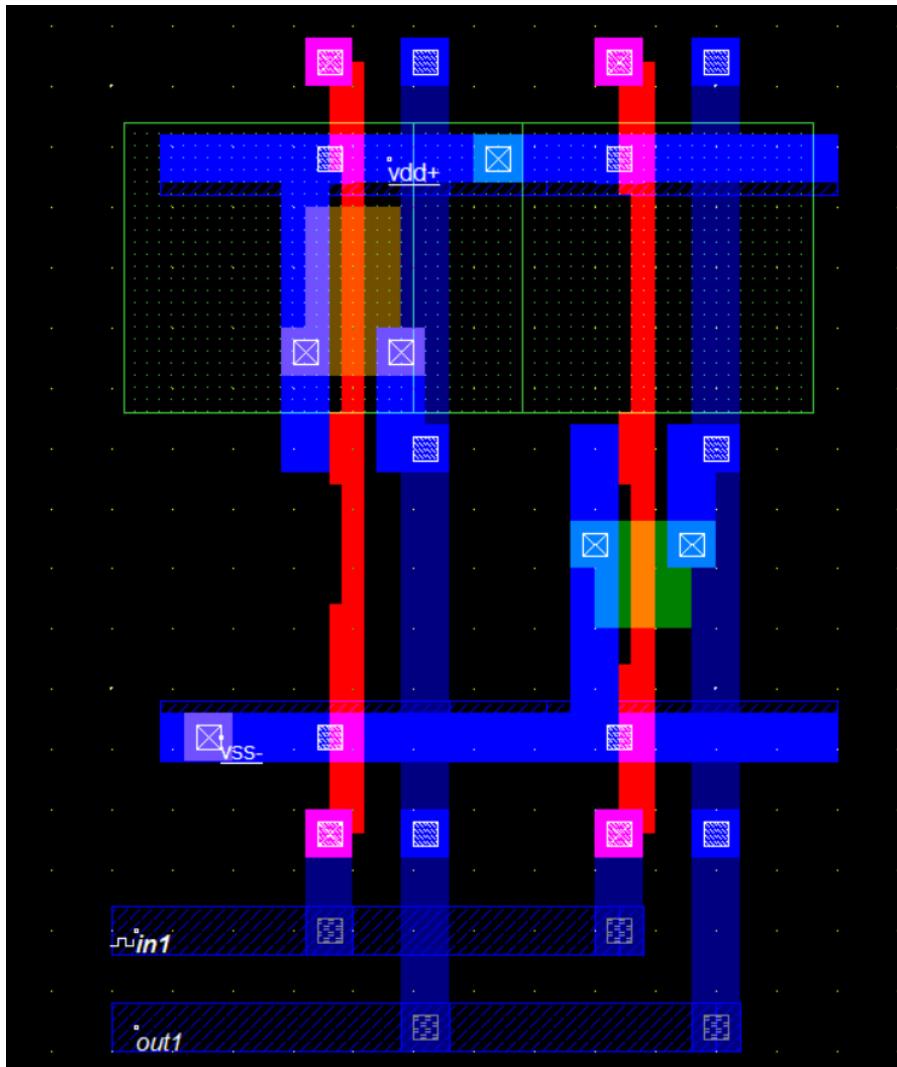
Misc. settings:

- Time scale: 1.00
- Max clocks: 16

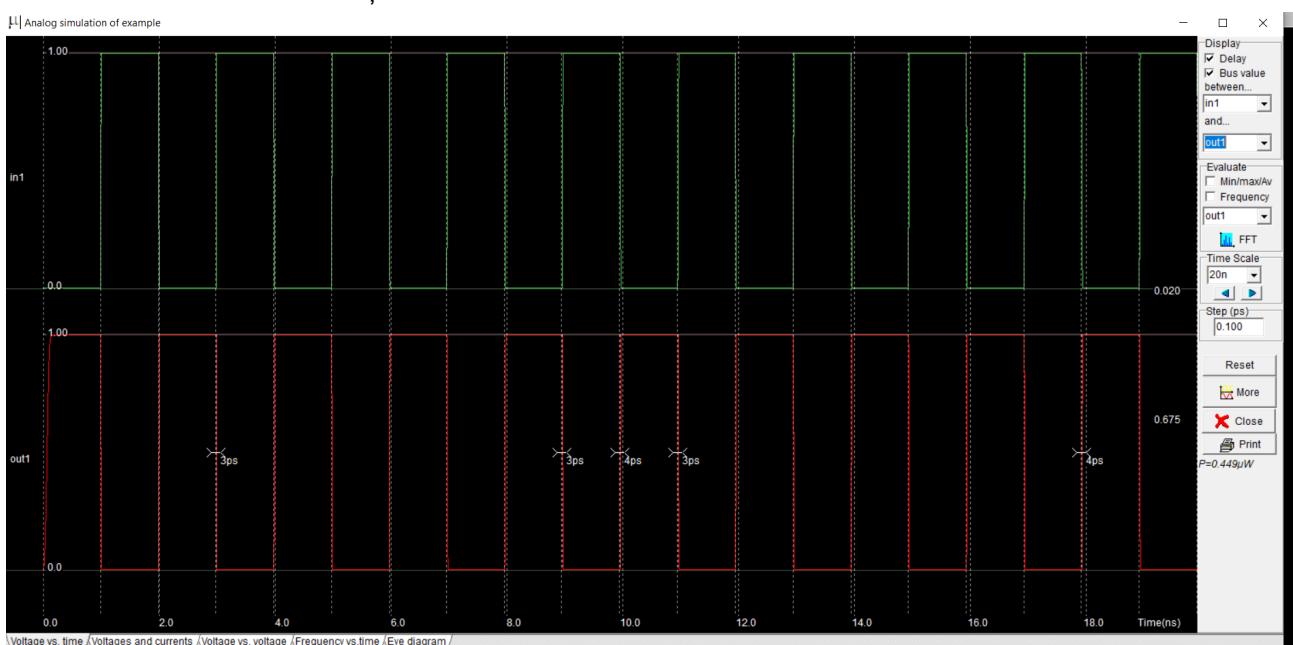
Buttons at the bottom:

- Update Verilog
- Extract circuit
- OK

e. Implementarea în Microwind

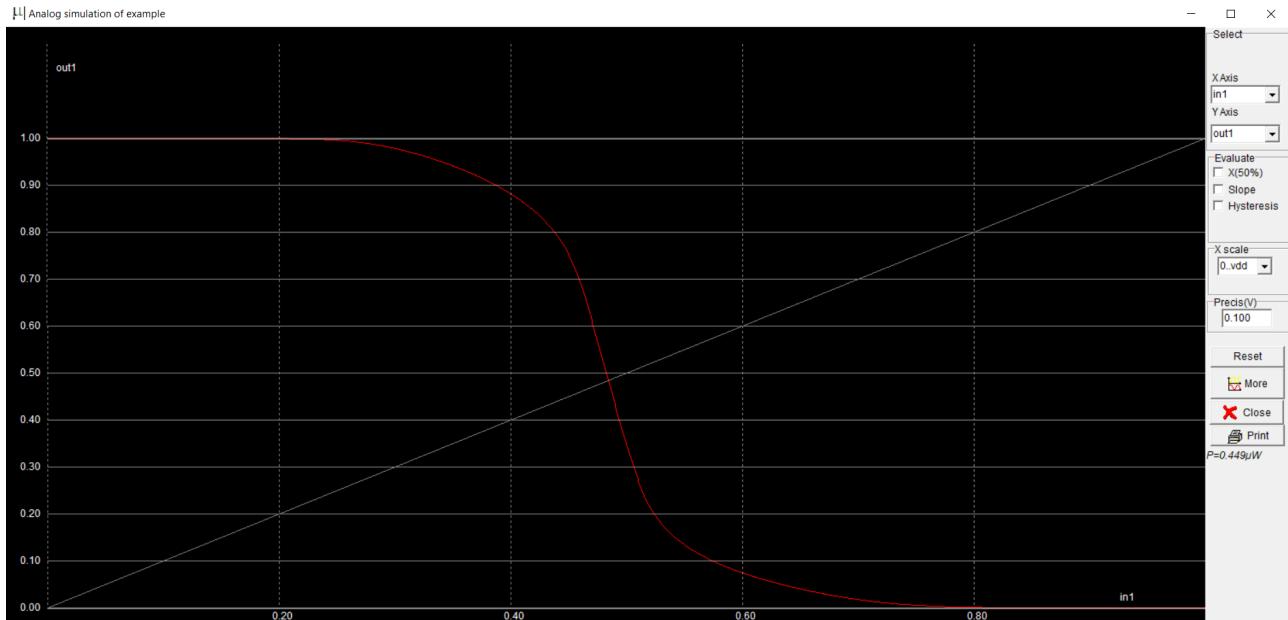


f. Simularea obținută în Microwind



Vin/t și Vout/t

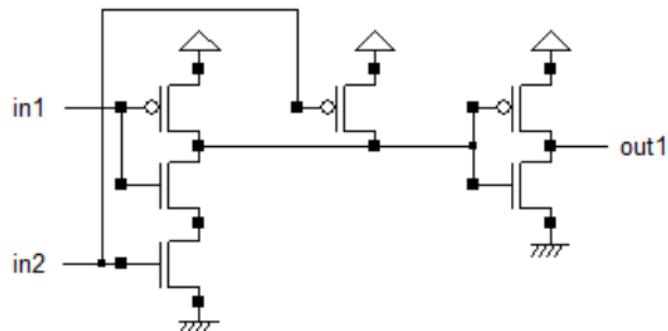
Analog simulation of example



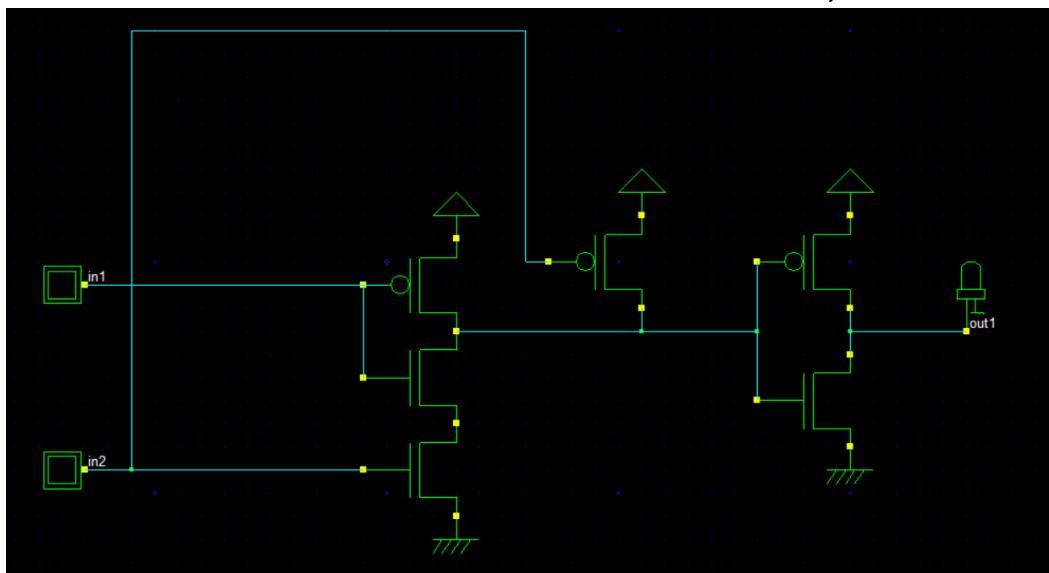
V_{out}/V_{in}

2. AND

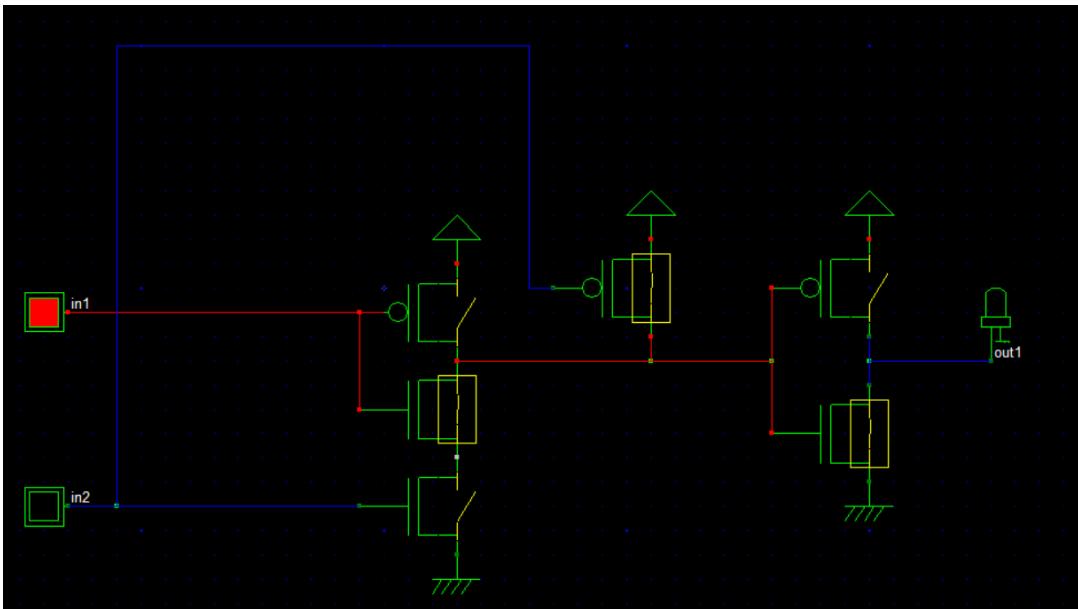
a. Schema circuitului



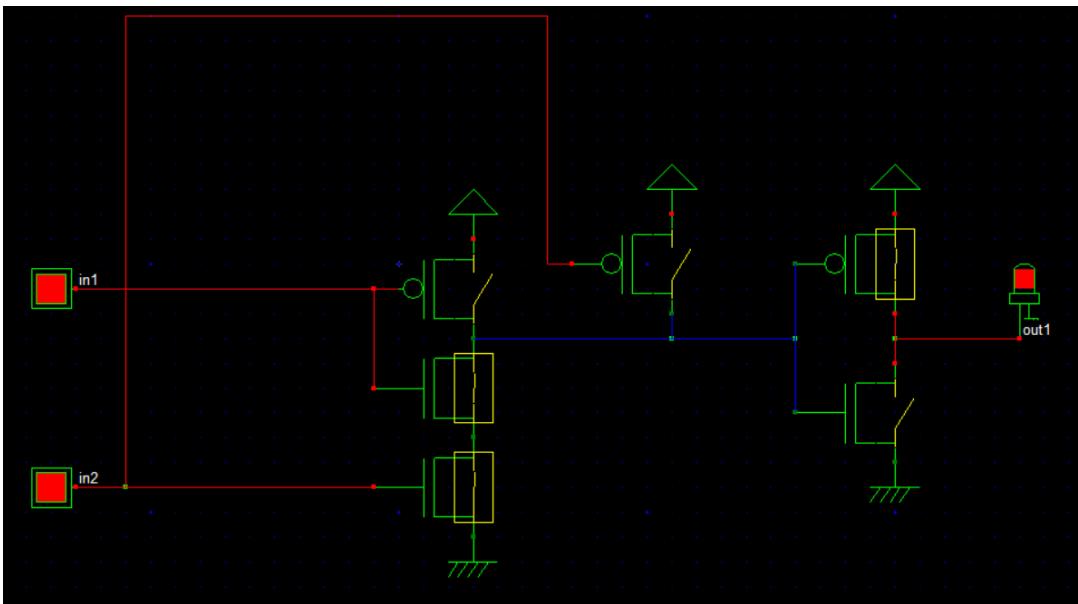
b. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS



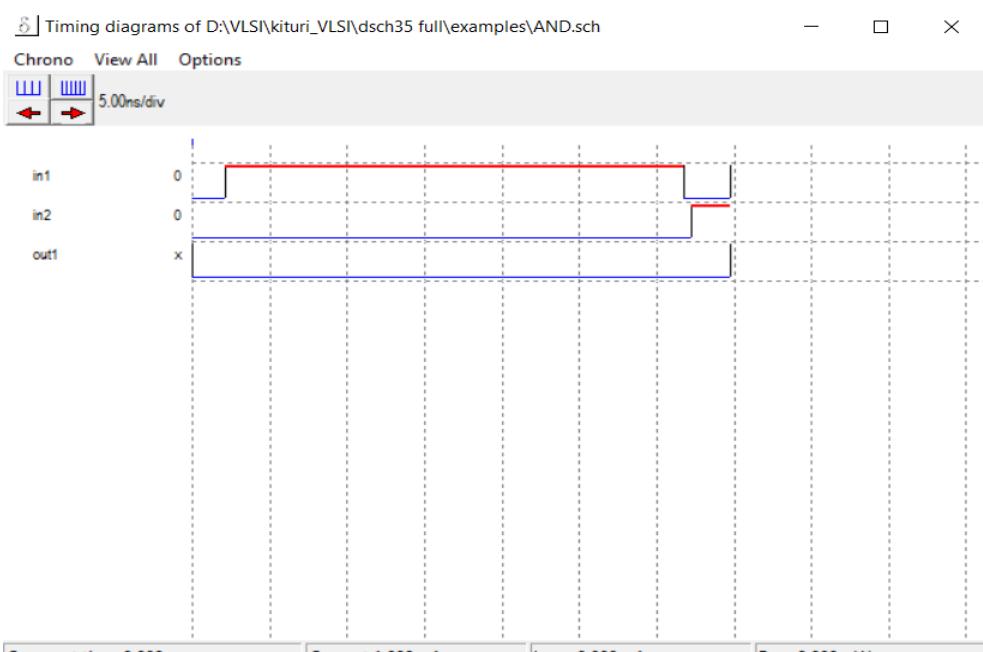
c. Simularea în DSCH



Input 1 0, Output 0



Input 1 1, Output 1



d. Codul Verilog obținut

The screenshot shows the DSCH 3.5 software interface. The top menu bar includes 'File', 'Edit', 'Verilog', 'Hierarchy', 'Netlist', 'Critical path', and 'Help'. The main window displays Verilog code for an AND gate. The code defines a module 'AND' with inputs 'in1', 'in2' and output 'out1'. It uses six transistors (pmos_1 to pmos_6) to implement the logic. Simulation parameters are also included. On the right side, there is an 'Information' panel with settings for module name ('AND'), and checkboxes for adding gate delay info, appending simulation information, and adding labels as comments. Below this, statistics are provided: 24 lines of Verilog code, 14 symbols in the design, and 6 nodes in the circuit. There are also options for adding line numbers and time scale/max clocks. At the bottom are buttons for 'Update Verilog', 'Extract circuit', and 'OK'.

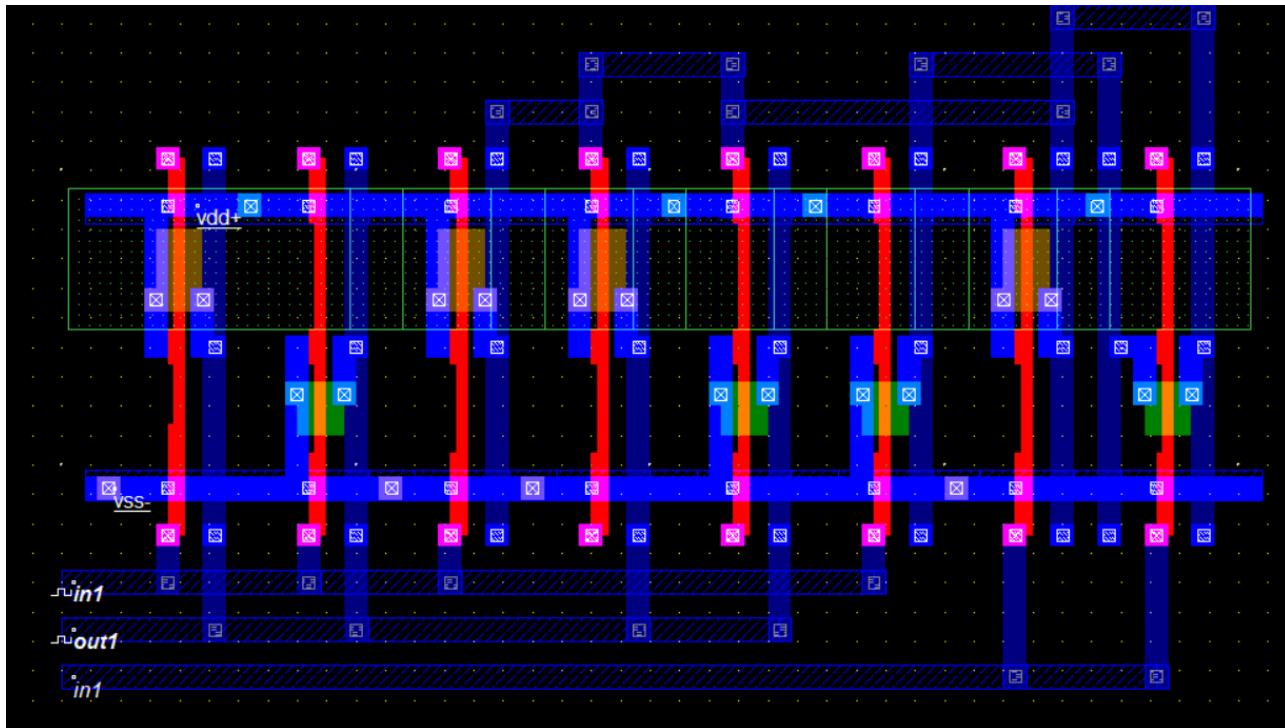
```
// DSCH 3.5
// 5/10/2023 11:26:21 PM
// D:\VLSI\kituri_VLSI\dsch35 full\examples\AND.sch

module AND( in1,in2,out1);
    input in1,in2;
    output out1;
    wire w4,w6;
    pmos #(3) pmos_1(w4,vdd,in2); // 0.5u 0.07u
    pmos #(2) pmos_2(out1,vdd,w4); // 0.5u 0.07u
    nmos #(2) nmos_3(out1,vss,w4); // 0.3u 0.07u
    nmos #(1) nmos_4(w6,vss,in2); // 0.3u 0.07u
    pmos #(3) pmos_5(w4,vdd,in1); // 0.5u 0.07u
    nmos #(3) nmos_6(w4,w6,in1); // 0.3u 0.07u
endmodule

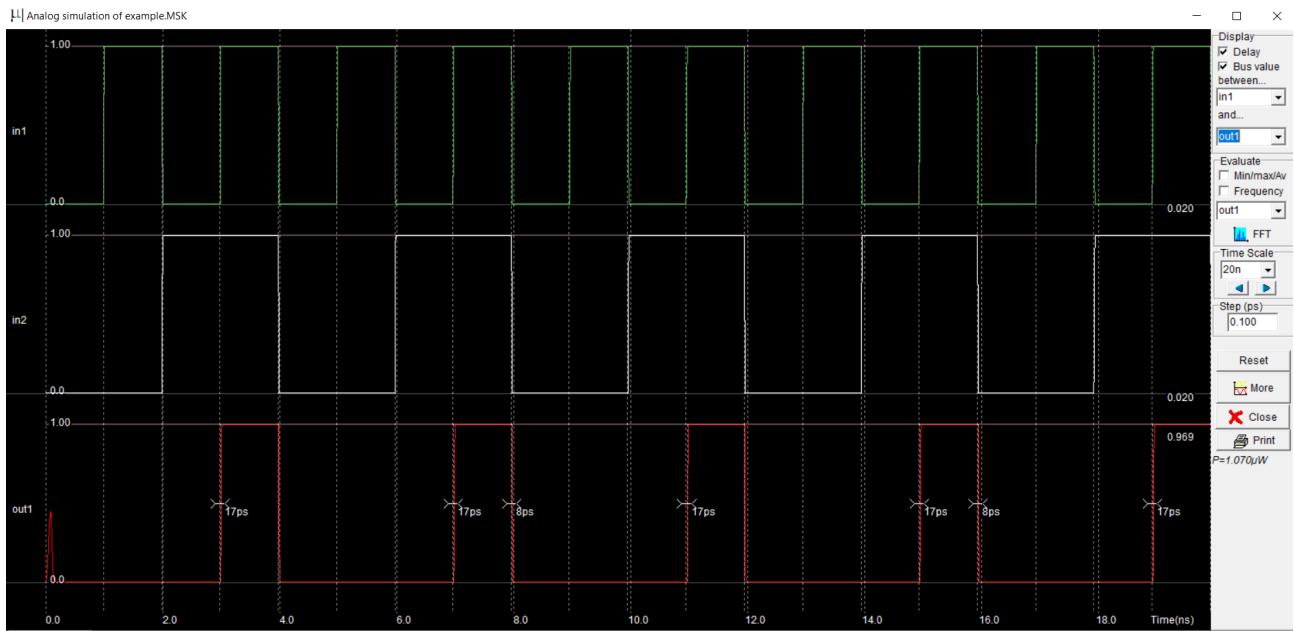
// Simulation parameters in Verilog Format
always
#200 in1=~in1;
#400 in2=~in2;

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

e. Implementarea în Microwind

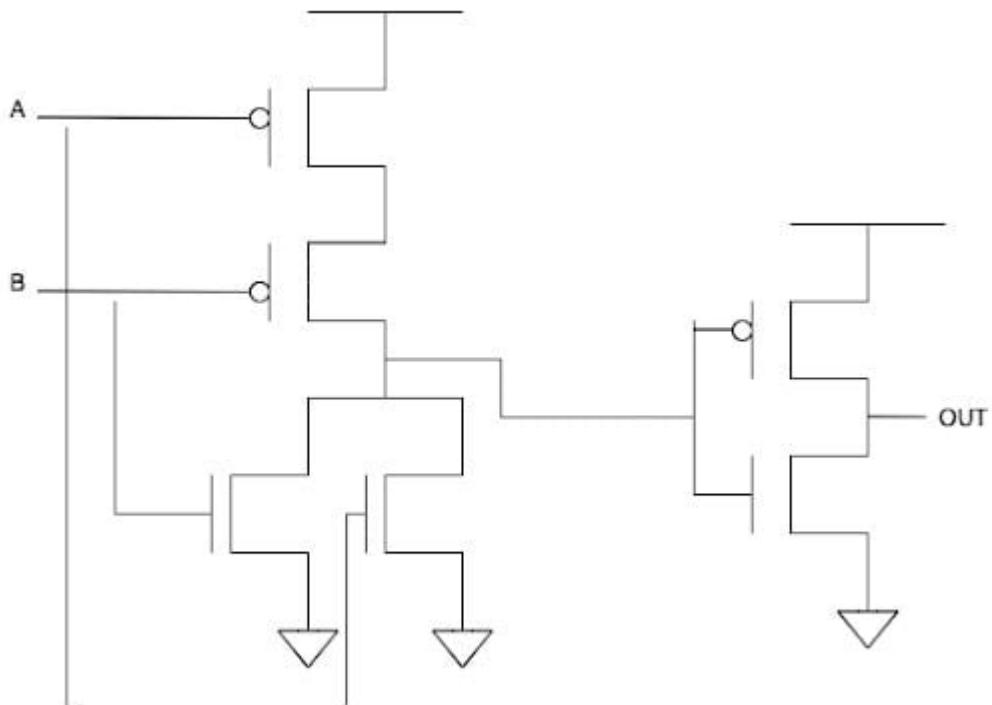


f. Simularea obținută în Microwind



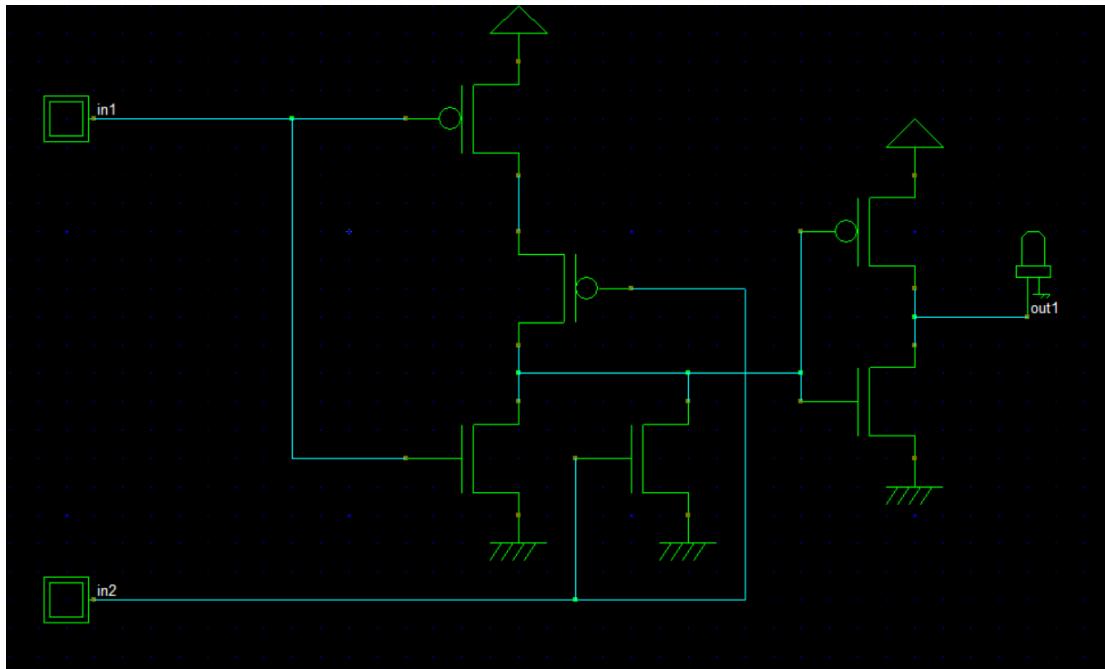
3. OR

a. Schema circuitului (sursa: <https://builtin.com/hardware/nmos-transistor>)



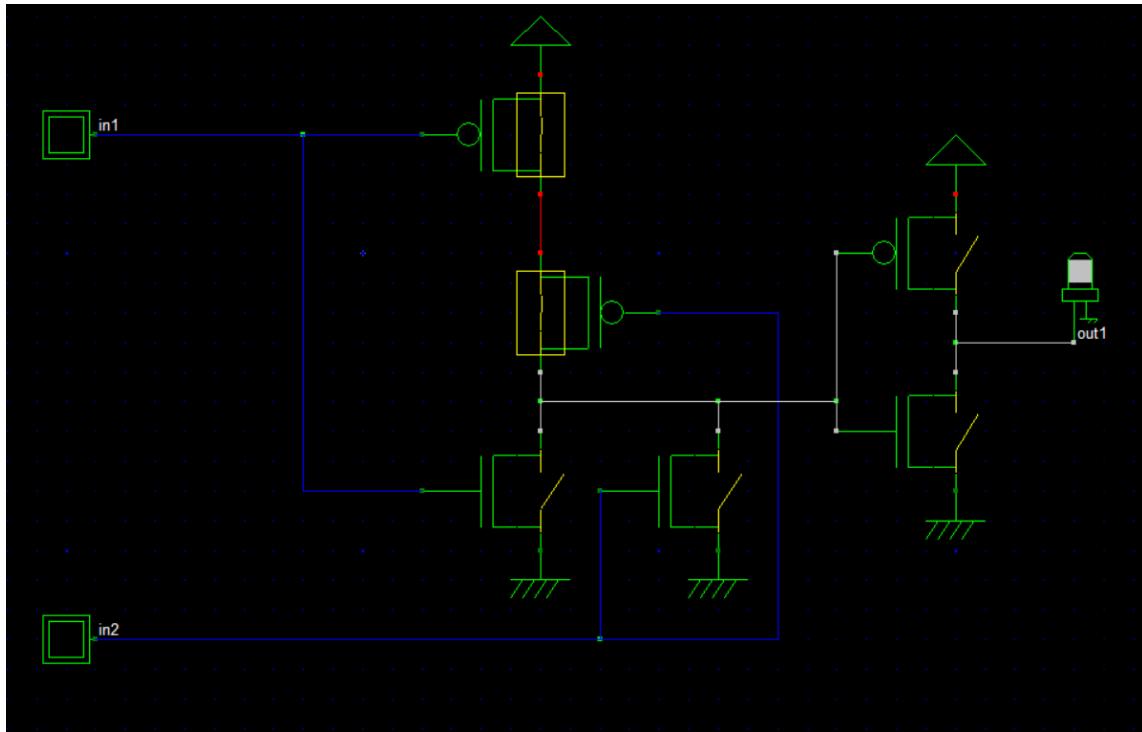
Example of an OR gate. | Image: Brendan Massey

b. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS

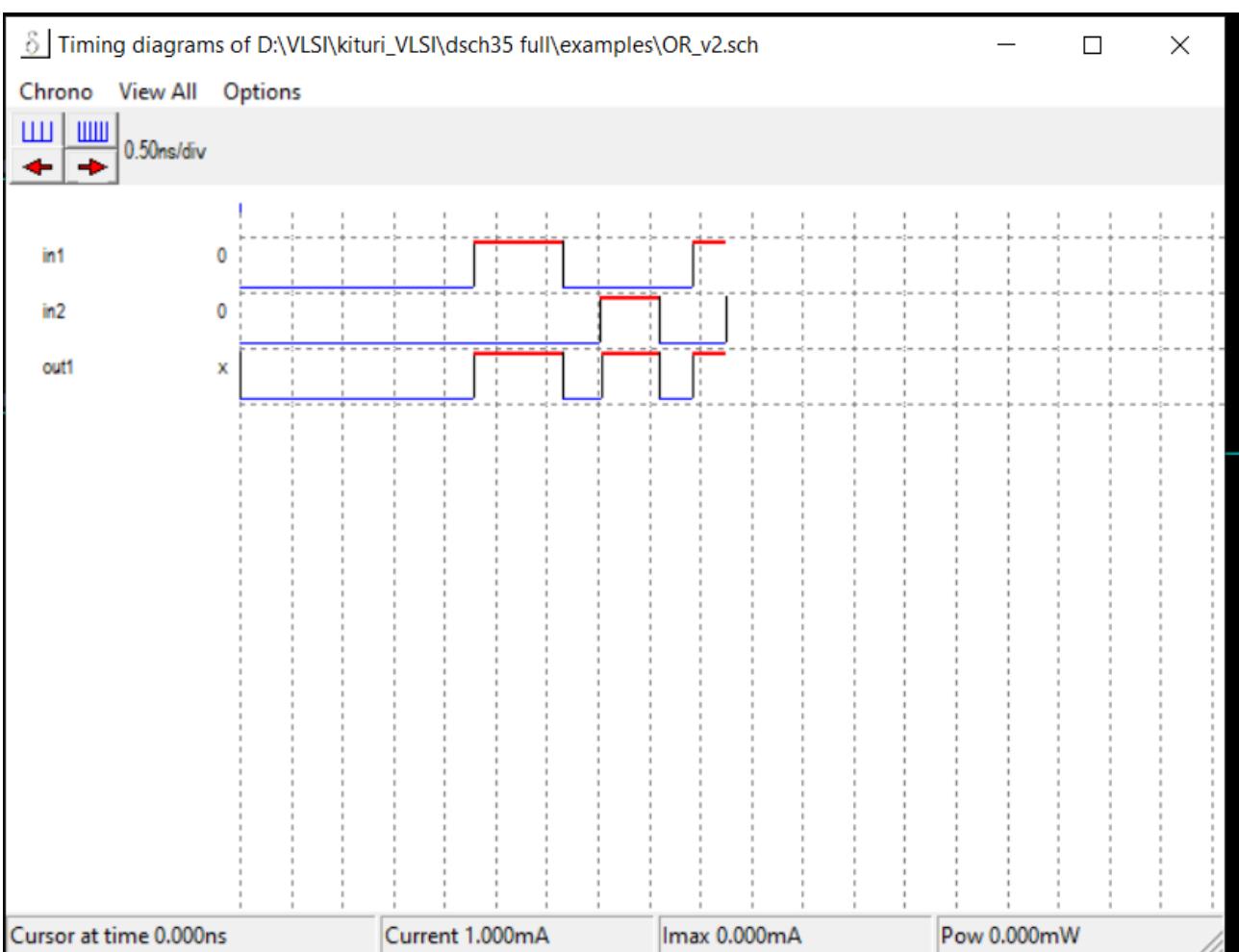
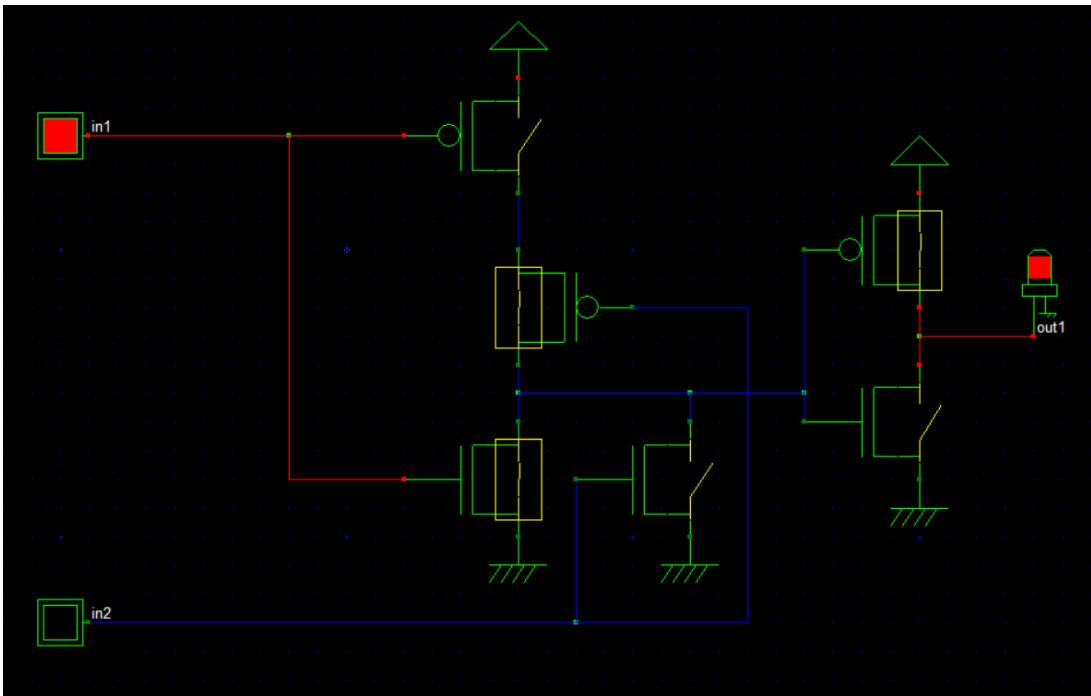


c. Simularea în DSCH

Input 0 0, Output 0



Input 1 0, Output 1



d. Codul Verilog obținut

Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path

```

// DSCH 3.5
// 5/11/2023 1:35:45 AM
// D:\VLSI\kituri_VLSI\dsch35 full\examples\OR_v3.sch

module OR_v3( in1,in2,out1);
    input in1,in2;
    output out1;
    wire w4,w6;
    nmos #(3) nmos_1(w4,vss,in2); // 0.3u 0.07u
    nmos #(2) nmos_2(out1,vss,w4); // 0.3u 0.07u
    pmos #(1) pmos_3(w6,vdd,in1); // 0.5u 0.07u
    pmos #(1) pmos_4(w6,w4,in2); // 0.5u 0.07u
    pmos #(2) pmos_5(out1,vdd,w4); // 0.5u 0.07u
    nmos #(3) nmos_6(w4,vss,in1); // 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

```

Information

Module name (8 char. max)
OR_v3

Add gate delay info
 Append simul. infomations
 Add labels as comments

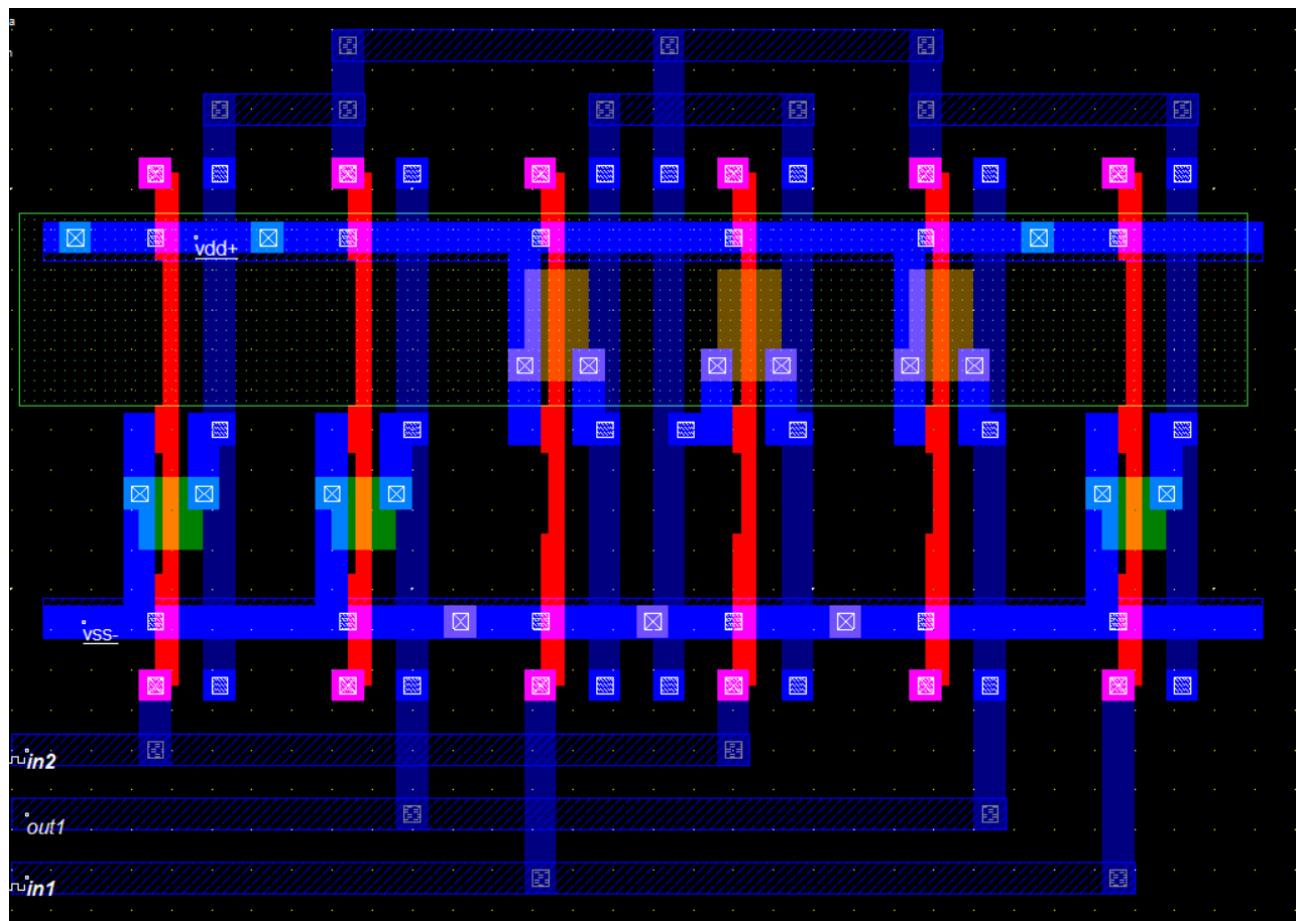
The Verilog file has 24 lines
The design includes 14 symbols
The circuit has 6 nodes

Add line numbers

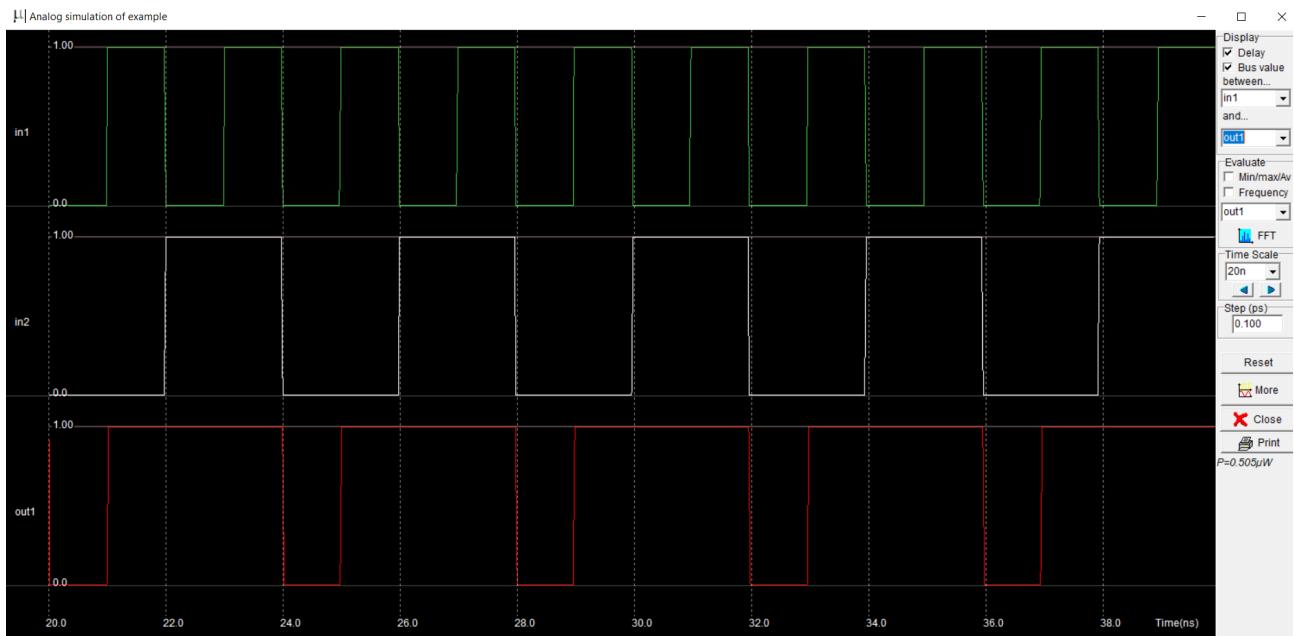
Misc.

Time scale : 1.00
Max clocks: 16

e. Implementarea în Microwind

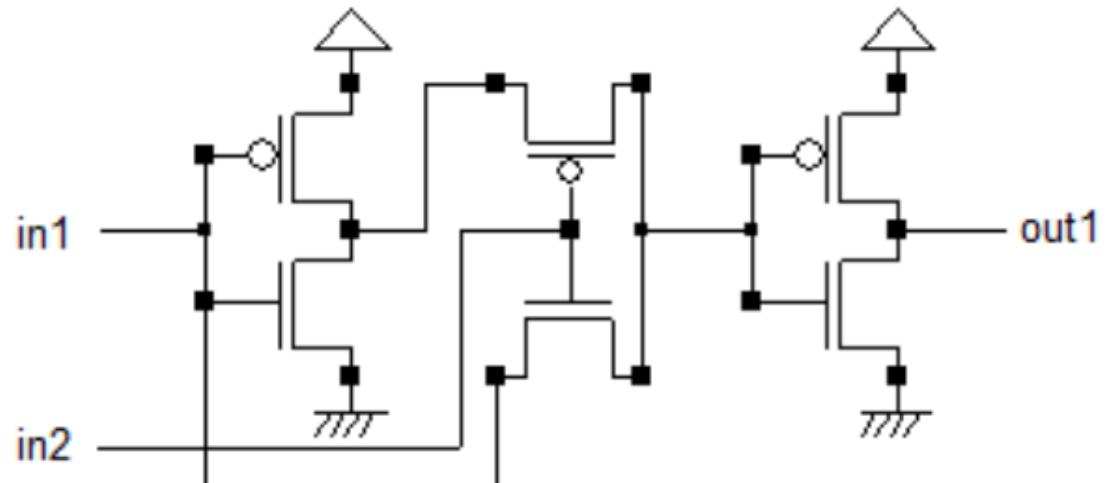


f. Simularea obținută în Microwind

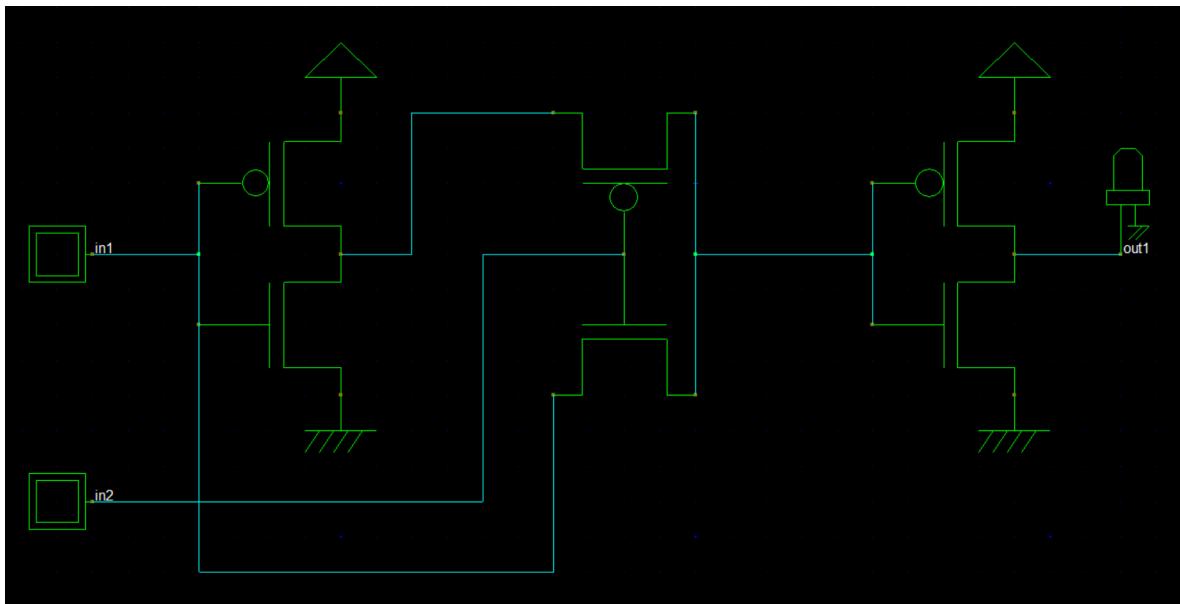


4. XOR

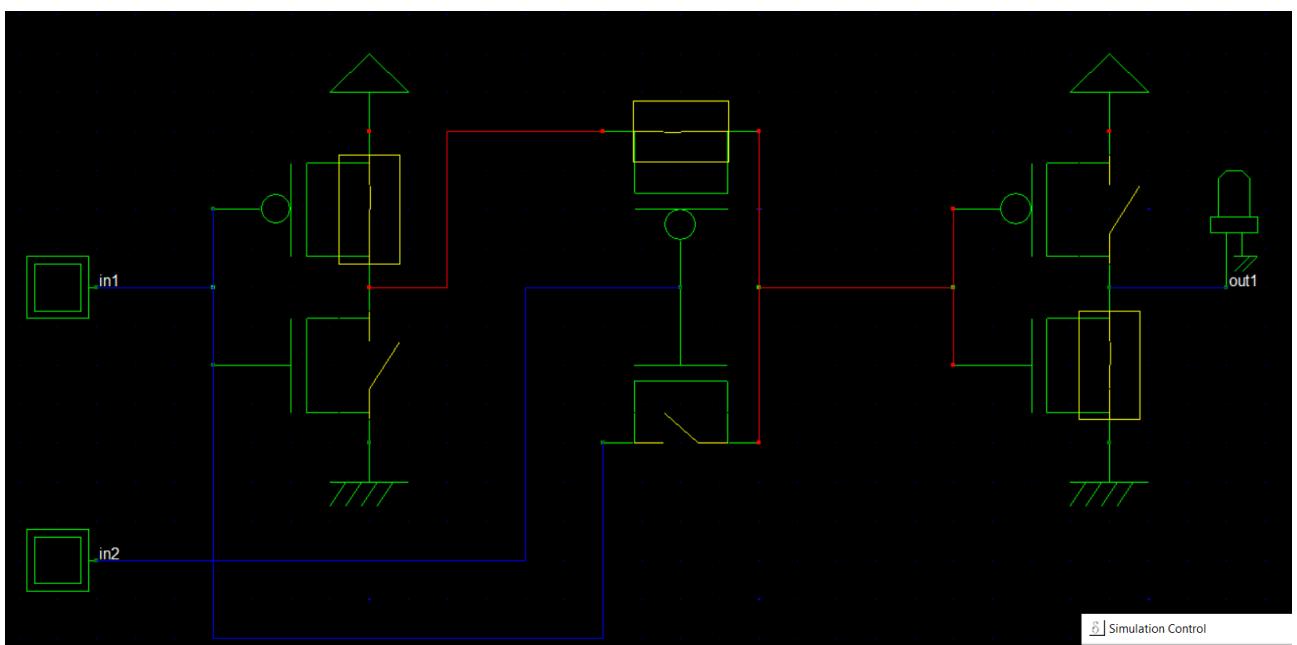
a. Schema circuitului (sursa: Cartea)



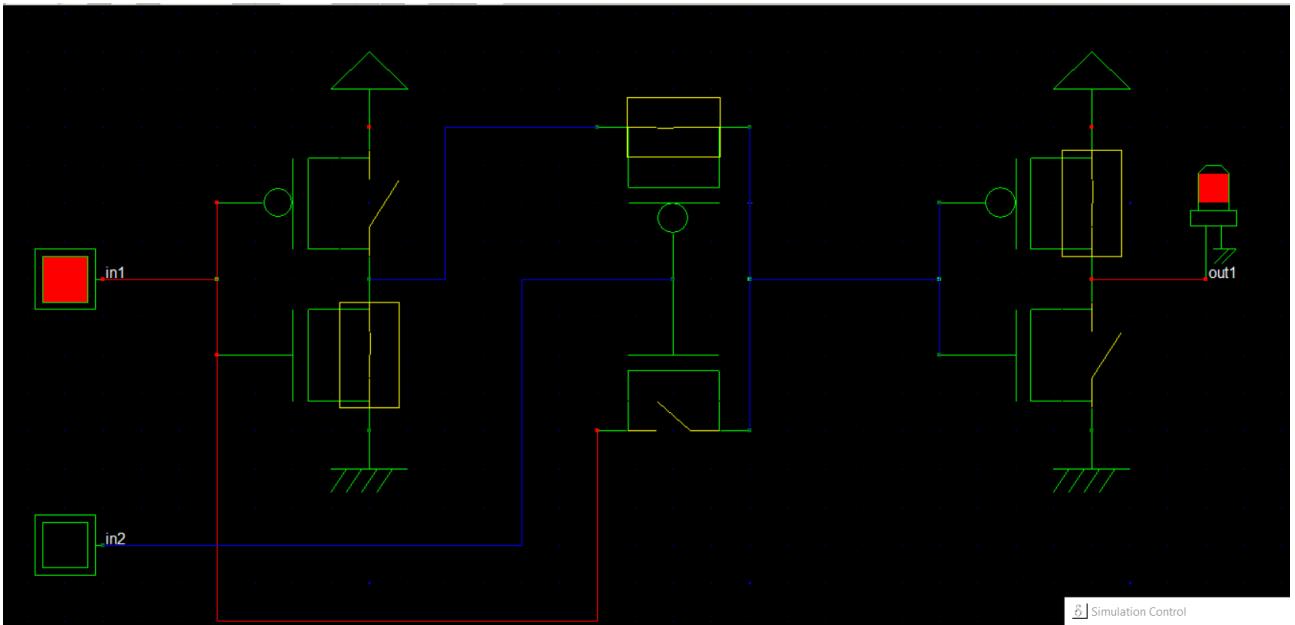
b. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS



c. Simularea în DSCH

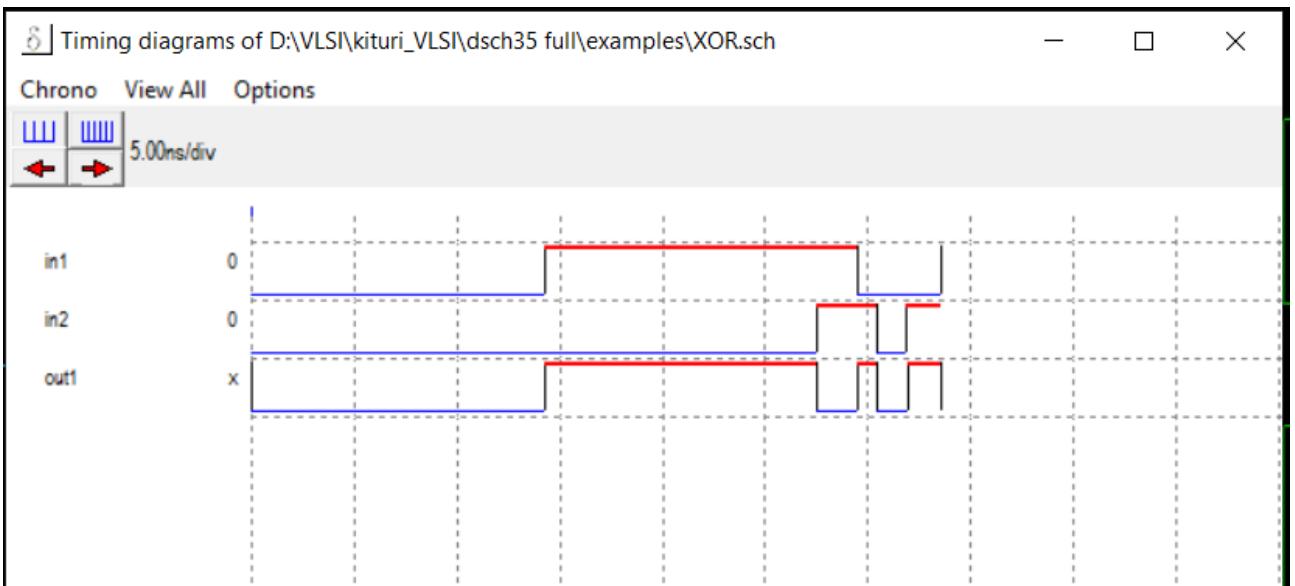


Input 0 0, Output 0



Simulation Control

Input 1 0, Output 1



Timing Diagram

d. Codul Verilog obținut

Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path |

```
// DSCH 3.5
// 5/11/2023 1:46:39 AM
// D:\VLSI\kituri_VLSI\dsch35_full\examples\XOR.sch

module XOR( in1,in2,out1);
    input in1,in2;
    output out1;
    wire w3,w4;
    pmos #(2) pmos_1(out1,vdd,w3); // 0.5u 0.07u
    nmos #(2) nmos_2(out1,vss,w3); // 0.3u 0.07u
    pmos #(2) pmos_3(w3,w4,in2); // 0.5u 0.07u
    nmos #(2) nmos_4(w3,in1,in2); // 0.3u 0.07u
    nmos #(2) nmos_5(w4,vss,in1); // 0.3u 0.07u
    pmos #(2) pmos_6(w4,vdd,in1); // 0.5u 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
```

Information

Module name (8 char. max)
XOR

Add gate delay info
 Append simul. infomations
 Add labels as comments

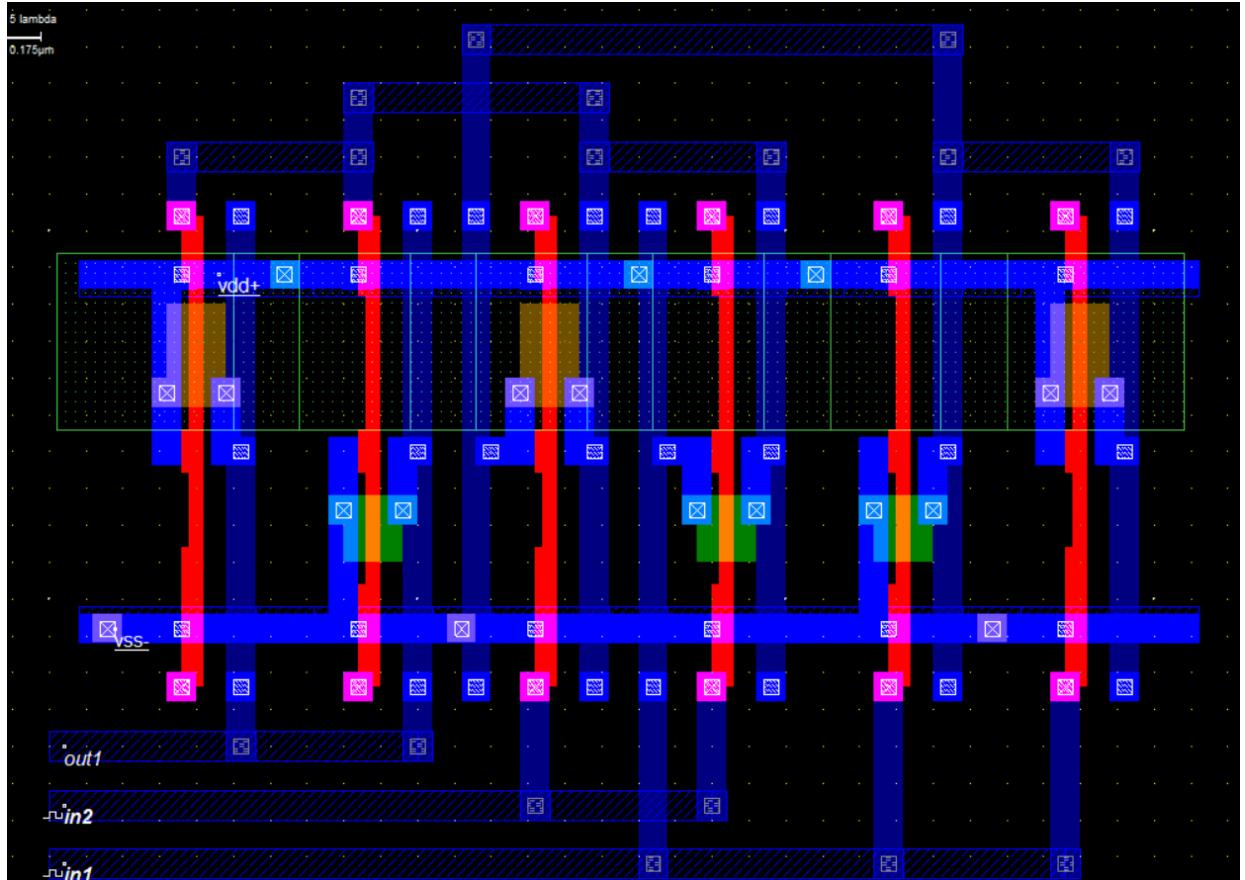
The Verilog file has 24 lines
The design includes 13 symbols
The circuit has 6 nodes

Add line numbers

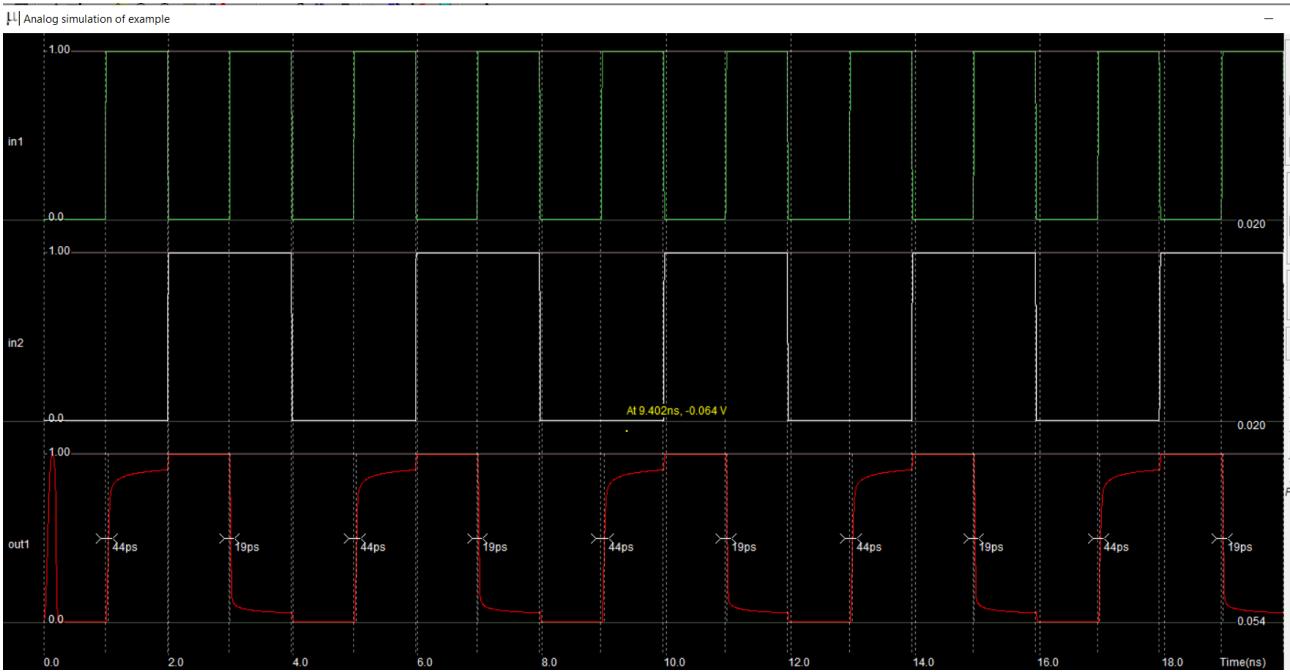
Misc.

Time scale : 1.00
Max clocks: 16

e. Implementarea în Microwind

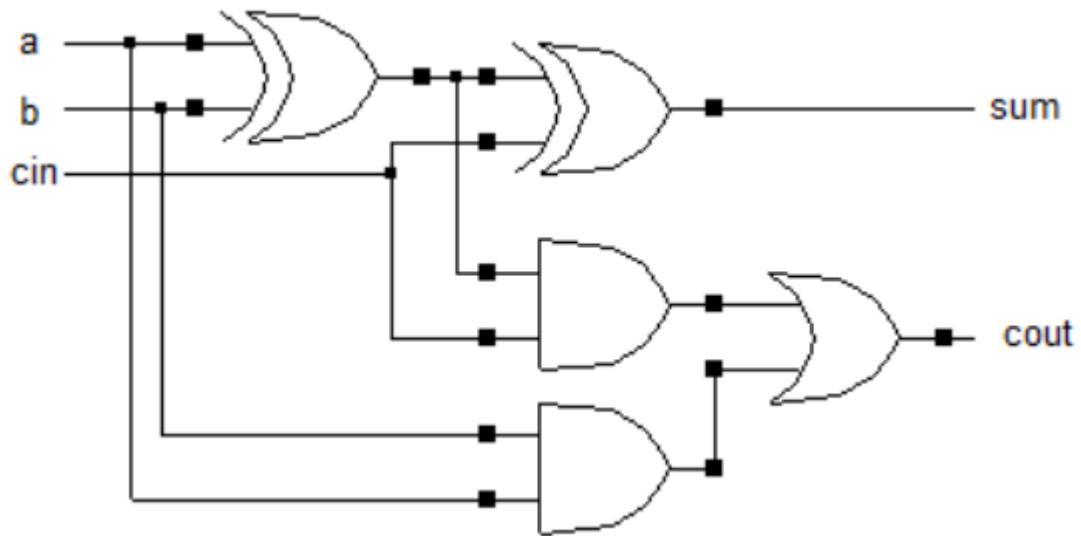


f. Simularea obținută în Microwind

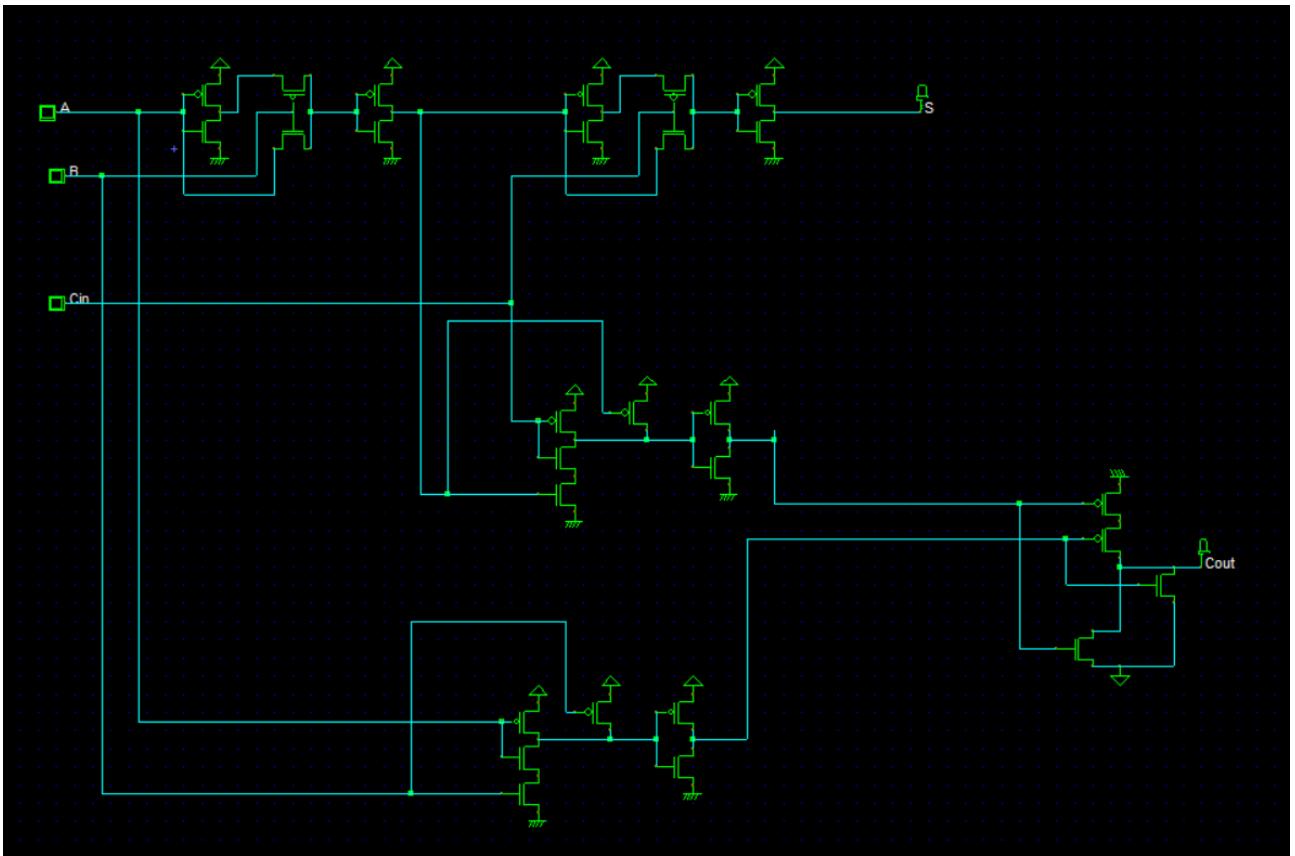


5. Sumator

a. Schema circuitului (sursa: Cartea)

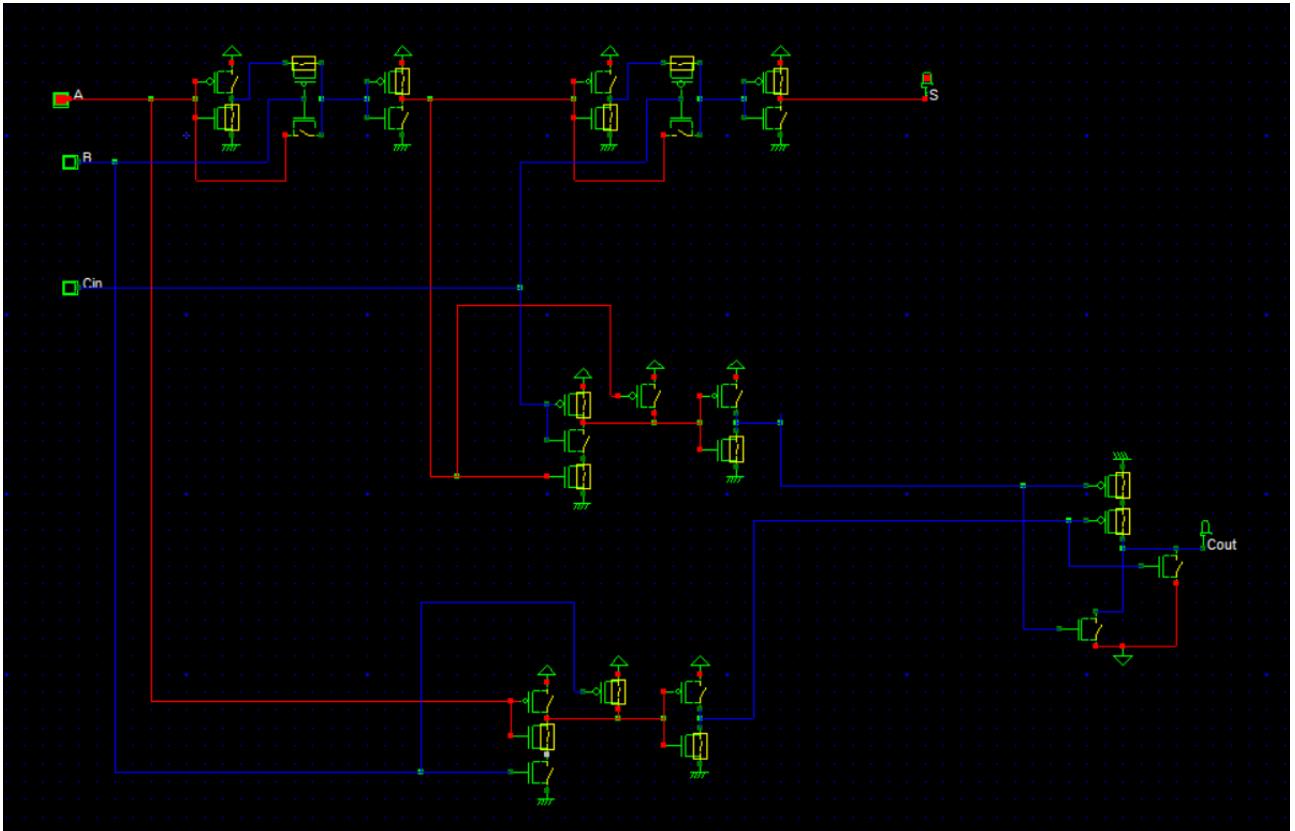


b. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS

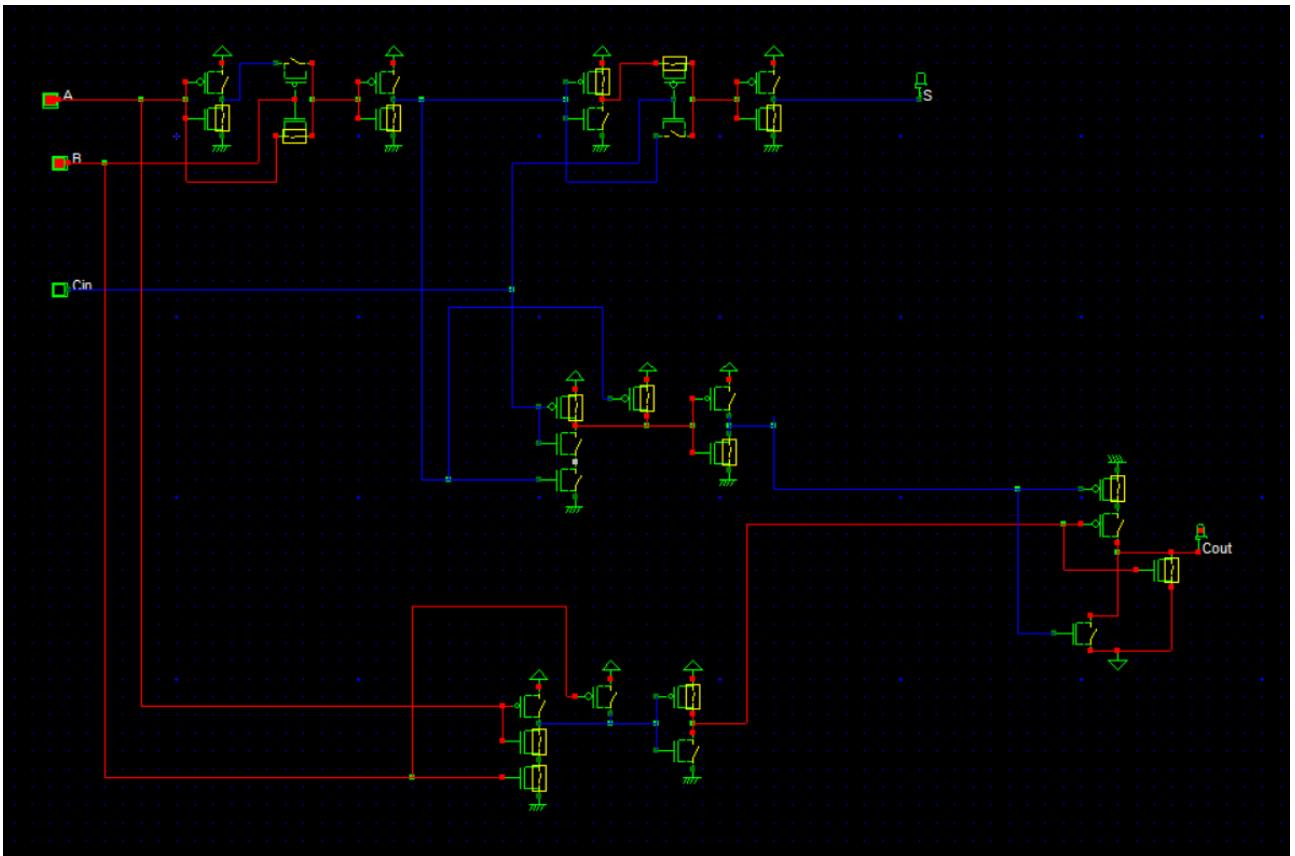


c. Simularea în DSCH

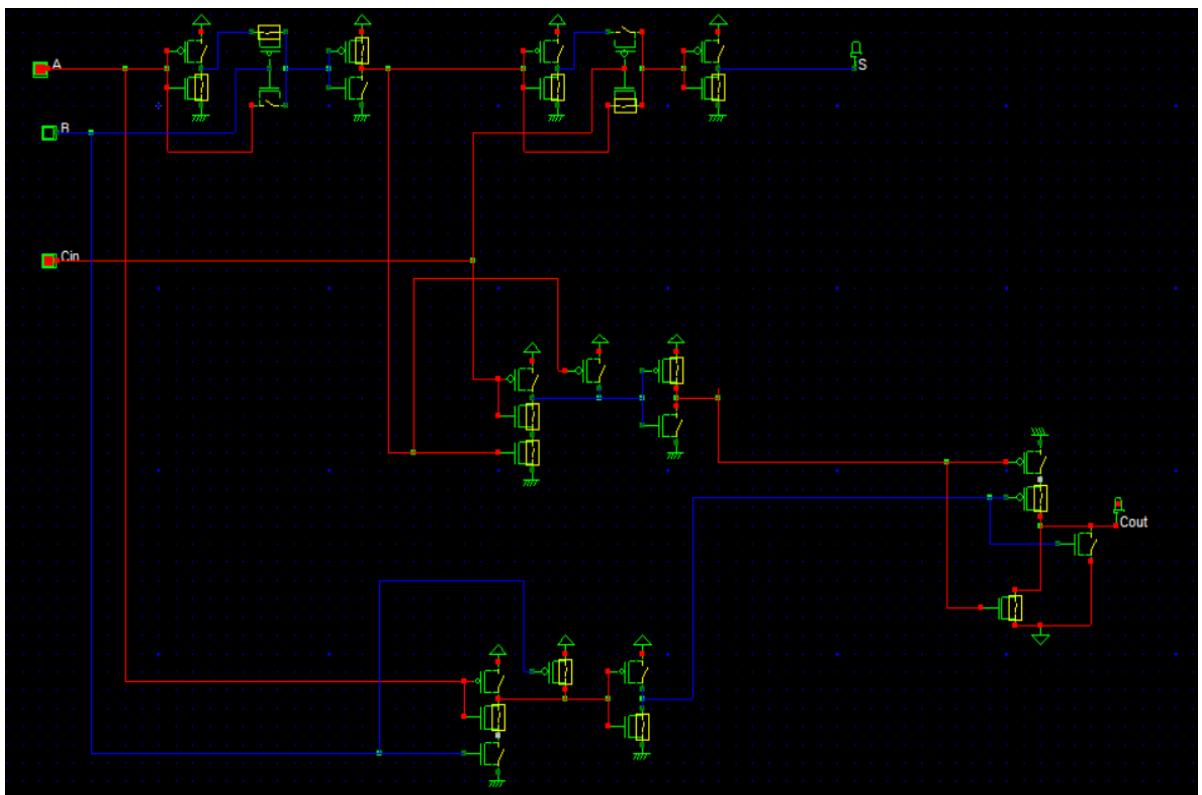
d. Codul Verilog obținut



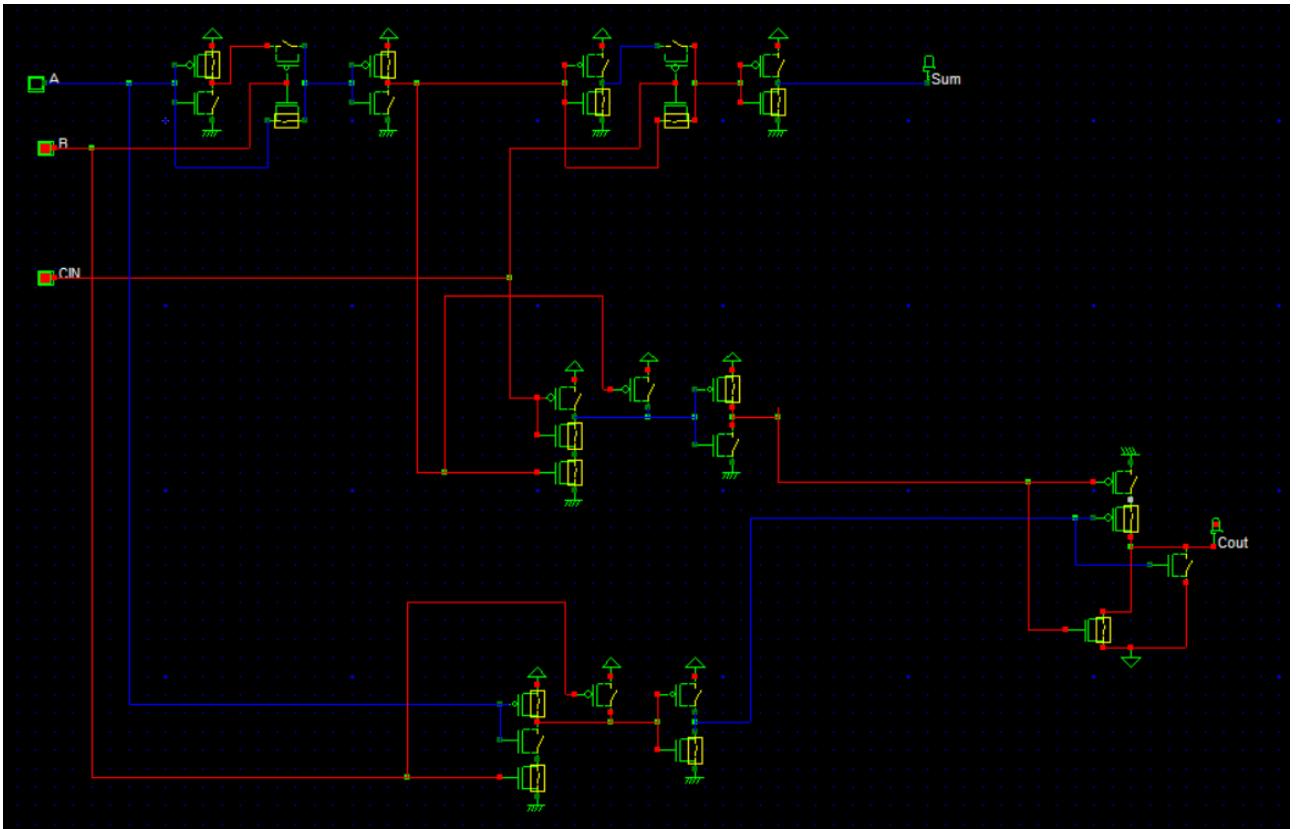
A = 1, B = 0, CIN = 0, COUNT = 0, SUM = 1



A = 1, B = 1, CIN = 0, COUNT = 1, SUM = 0



A = 1, B = 0, CIN = 1, COUNT = 1, SUM = 0



A = 0, B = 1, CIN = 1, COUNT = 1, SUM = 0

d. Codul Verilog obținut

Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path

```
// DSCH 3.5
// 5/11/2023 2:32:03 AM
// D:\VLSI\kituri_VLSI\dsch35_full\examples\sumator_simplu.v

module sumator_simplu( B,Cin,A,S,Cout);
    input B,Cin,A;
    output S,Cout;
    wire w4,w7,w8,w9,w11,w12,w13,w14;
    wire w15,w16,w17,w18;
    nmos #(2) nmos_1(Cout,vdd,w4); // 0.3u 0.07u
    pmos #(2) pmos_2(w7,vdd,A); // 0.5u 0.07u
    nmos #(2) nmos_3(w7,vss,A); // 0.3u 0.07u
    nmos #(2) nmos_4(w8,A,B); // 0.3u 0.07u
    pmos #(2) pmos_5(w8,w7,B); // 0.5u 0.07u
    nmos #(3) nmos_6(w9,vss,w8); // 0.3u 0.07u
    pmos #(3) pmos_7(w9,vdd,w8); // 0.5u 0.07u
    pmos #(2) pmos_8(S,vdd,w11); // 0.5u 0.07u
    nmos #(2) nmos_9(S,vss,w11); // 0.3u 0.07u
    pmos #(2) pmos_10(w11,w12,Cin); // 0.5u 0.07u
    nmos #(2) nmos_11(w11,w9,Cin); // 0.3u 0.07u
    nmos #(2) nmos_12(w12,vss,w9); // 0.3u 0.07u
    pmos #(2) pmos_13(w12,vdd,w9); // 0.5u 0.07u
    pmos #(2) pmos_14(Cout,w13,w14); // 0.5u 0.07u
    pmos #(3) pmos_15(w15,vdd,A); // 0.5u 0.07u
    pmos #(3) pmos_16(w16,vdd,w9); // 0.5u 0.07u
    pmos #(2) pmos_17(w4,vdd,w16); // 0.5u 0.07u
    nmos #(2) nmos_18(w4,vss,w16); // 0.3u 0.07u
    pmos #(1) pmos_19(w13,vss,w4); // 0.5u 0.07u
    nmos #(1) nmos_20(w17,vss,w9); // 0.3u 0.07u
    nmos #(3) nmos_21(w15,w18,A); // 0.3u 0.07u
    pmos #(3) pmos_22(w16,vdd,Cin); // 0.5u 0.07u
    nmos #(3) nmos_23(w16,w17,Cin); // 0.3u 0.07u
    pmos #(3) pmos_24(w15,vdd,B); // 0.5u 0.07u
    pmos #(2) pmos_25(w14,vdd,w15); // 0.5u 0.07u
    nmos #(2) nmos_26(w14,vss,w15); // 0.3u 0.07u
    nmos #(2) nmos_27(Cout,vdd,w14); // 0.3u 0.07u
    nmos #(1) nmos_28(w18,vss,B); // 0.3u 0.07u
endmodule

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

Information

Module name (8 char. max)
sumator_simplu
 Add gate delay info
 Append simul. infomations
 Add labels as comments

The Verilog file has 49 lines
The design includes 53 symbols
The circuit has 18 nodes

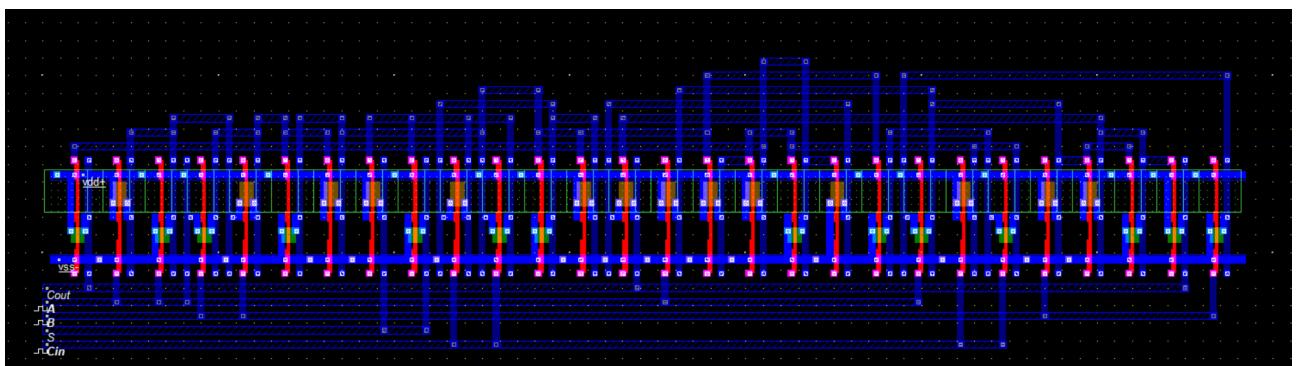
Add line numbers

Misc.

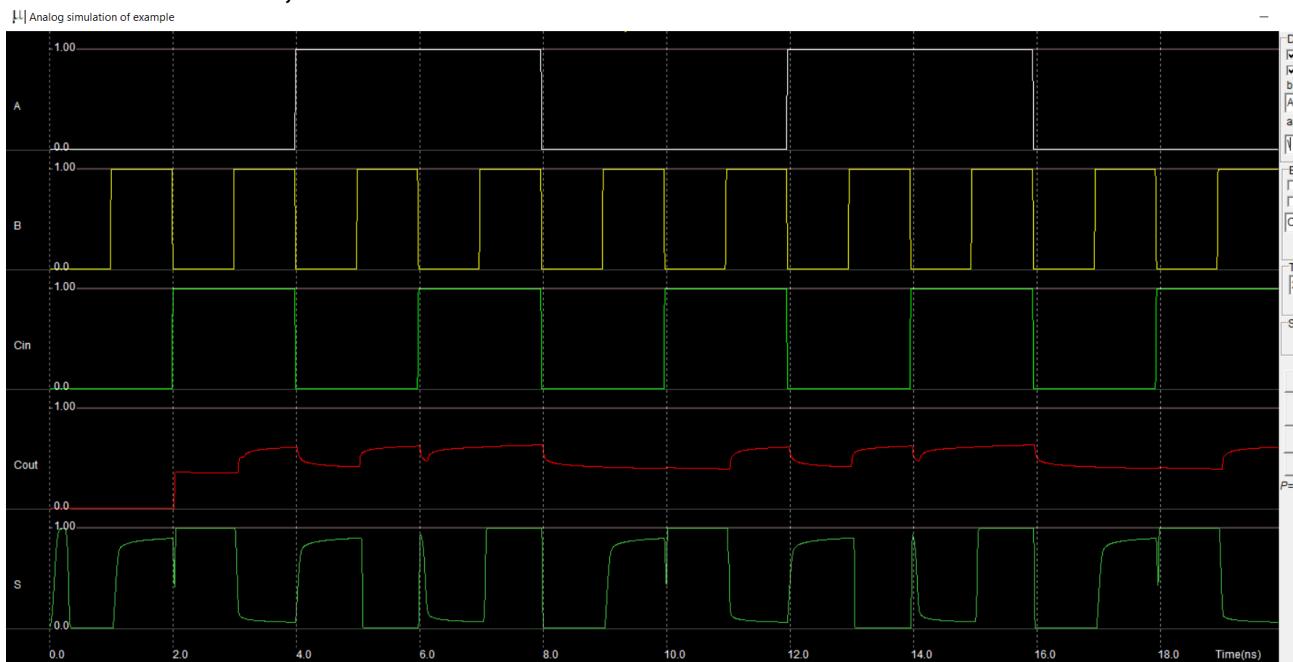
Time scale : 1.00
Max clocks: 16

[Update Verilog](#) [Extract circuit](#)

e. Implementarea în Microwind



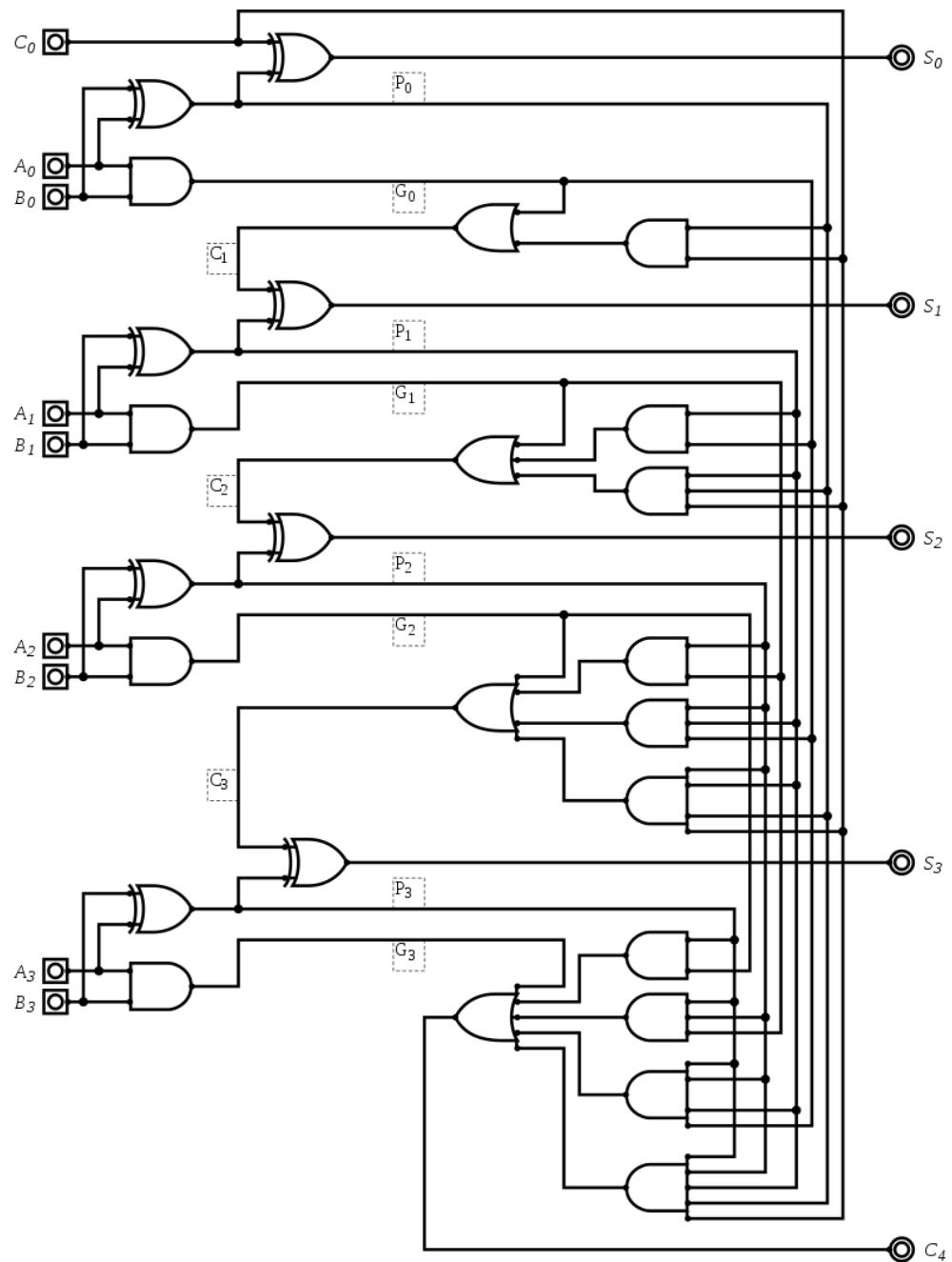
f. Simularea obținută în Microwind



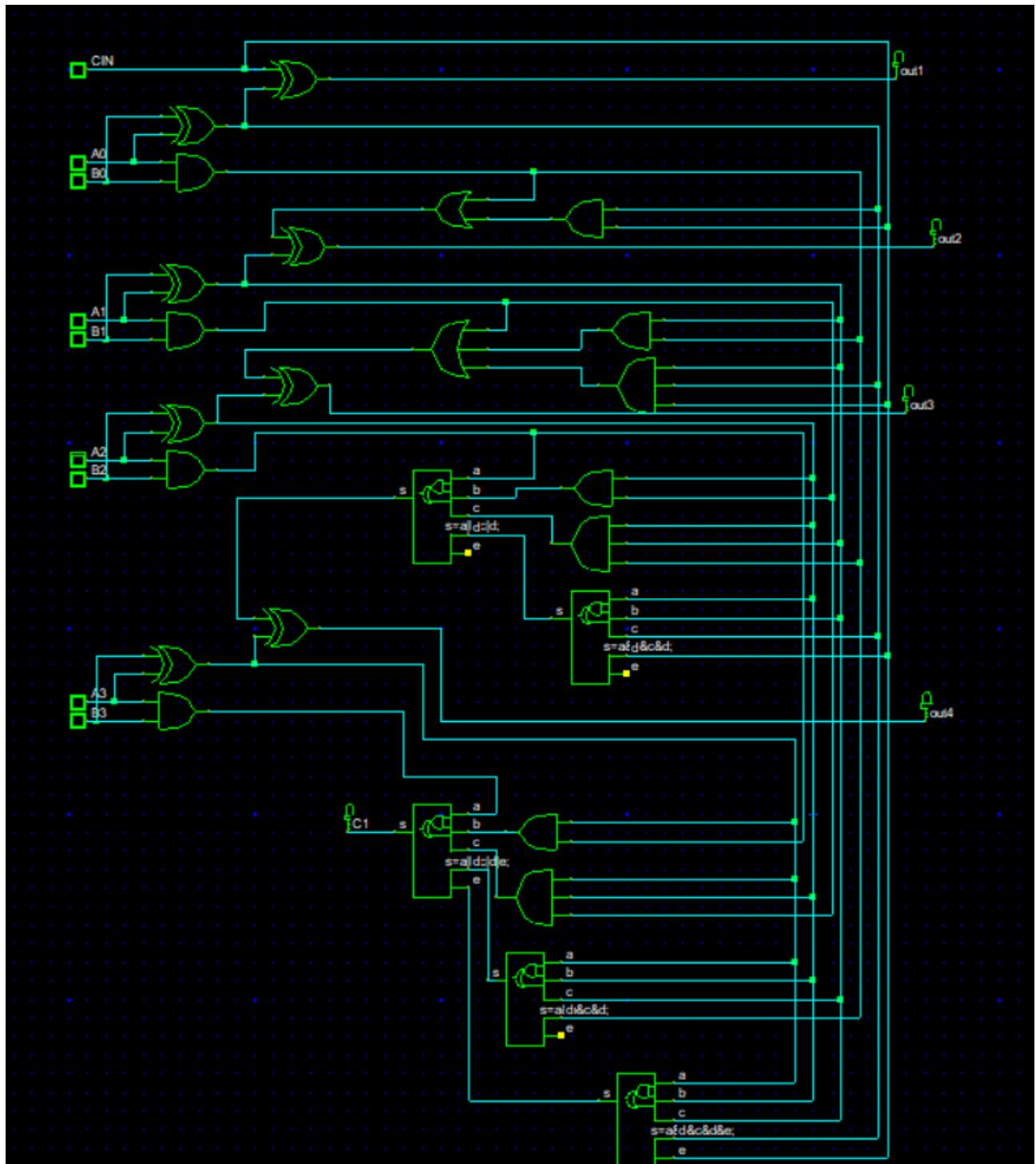
6. Sumator 4 biți Carry Look Ahead

A. Schema circuitului (sursa:

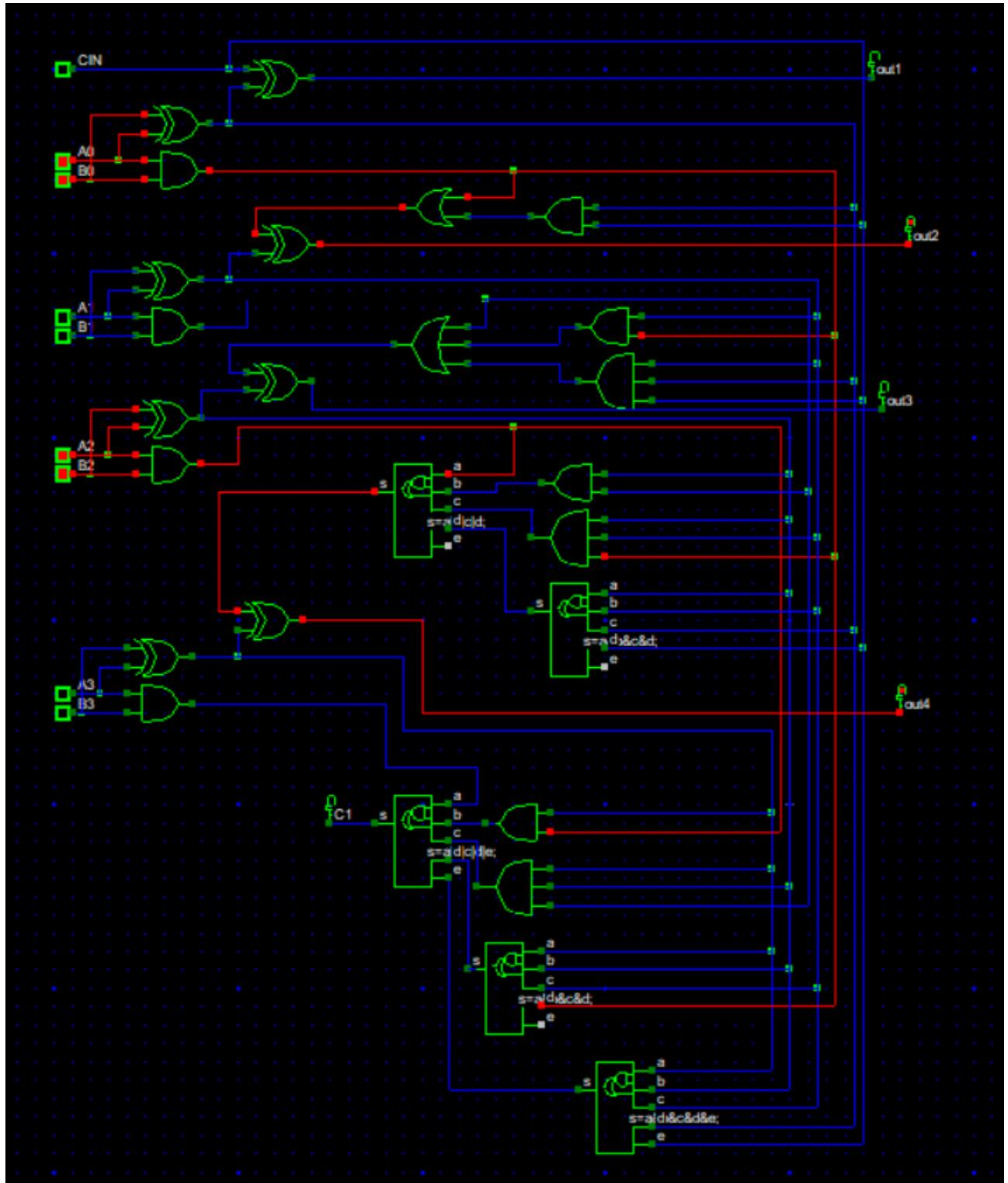
https://en.wikipedia.org/wiki/Carry-lookahead_adder#/media/File:Four_bit_adder_with_carry_lookahead.svg)



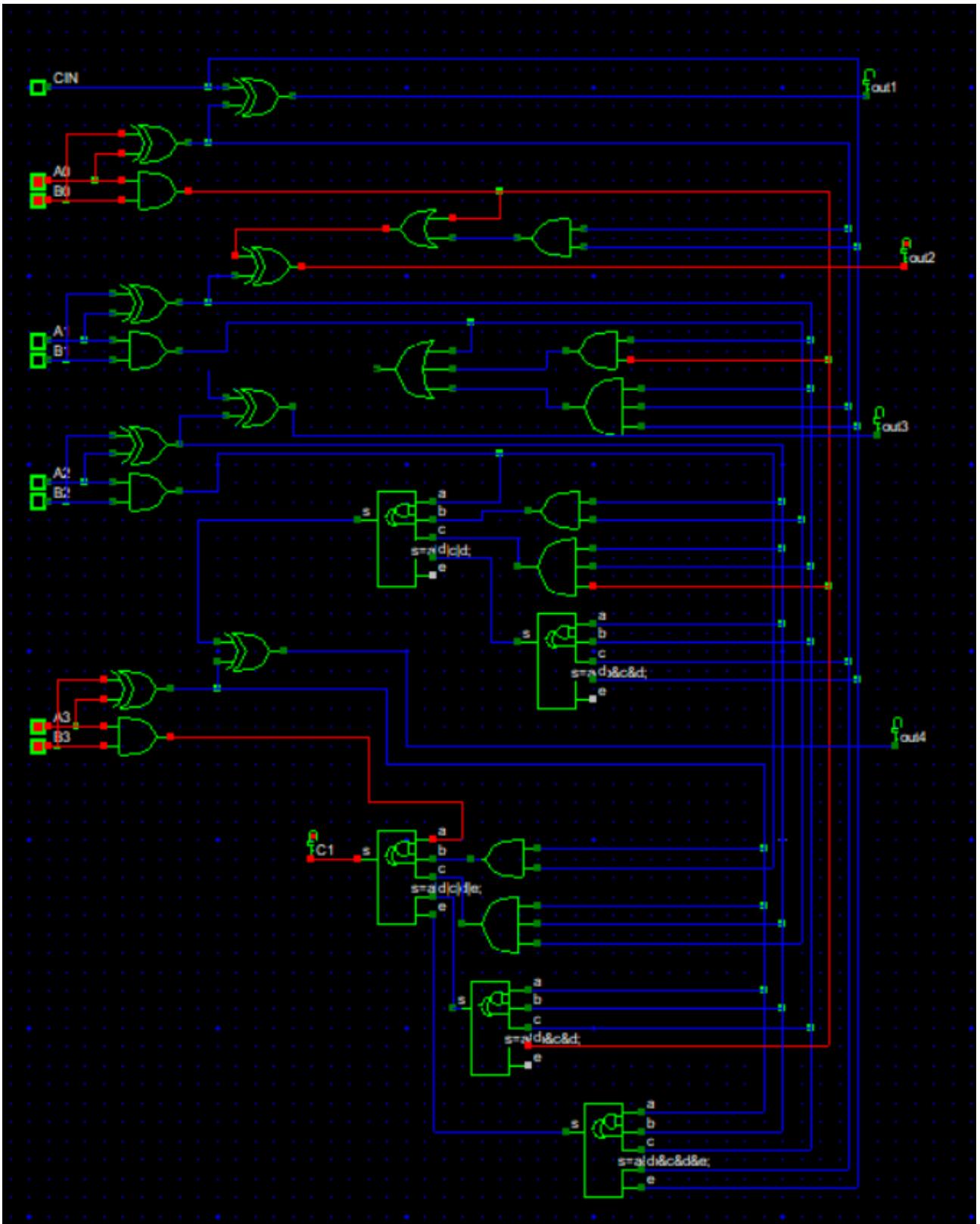
B. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS



C. Simularea în DSCH



$0101 + 0101$ cu $out3 = 1$ și $out1 = 1$, $c1 = 0$



1001 + 1001, out1 = 1, carry(c1) = 1

D. Codul Verilog obținut

Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path

```
// DSCH 3.5
// 5/11/2023 4:59:20 PM
// D:\VLSI\kituri_VLSI\dsch35 full\examples\sumator_carry_look_ahead.sch

module sumator_carry_look_ahead( CIN,A0,B0,A1,B1,A2,B2,A3,
B3,out1,out2,out3,out4,C1);
input CIN,A0,B0,A1,B1,A2,B2,A3;
input B3;
output out1,out2,out3,out4,C1;
wire w5,w6,w10,w11,w12,w14,w17,w18;
wire w19,w21,w22,w25,w26,w27,w29,w30;
wire w31,w32,w33,w34,w35,w36,w37,w39;
and #(5) and2_1(w5,B0,A0);
xor #(5) xor2_2(w6,B0,A0);
xor #(3) xor2_3(out1,CIN,w6);
and #(4) and2_4(w10,B1,A1);
xor #(6) xor2_5(w11,B1,A1);
xor #(3) xor2_6(out2,w12,w11);
or #(3) or2_7(w12,w14,w5);
and #(3) and2_8(w14,w6,CIN);
and #(4) and2_9(w17,B2,A2);
xor #(6) xor2_10(w18,B2,A2);
xor #(3) xor2_11(out3,w19,w18);
or #(4) or3_12(w19,w21,w22,w10);
and #(3) and2_13(w22,w5,w11);
and #(3) and3_14(w21,w11,w6,CIN);
and #(3) and2_15(w25,B3,A3);
xor #(5) xor2_16(w26,B3,A3);
xor #(3) xor2_17(out4,w27,w26);
assign w27=w17|w29|w30|w31;
and #(3) and2_18(w29,w10,w18);
and #(3) and3_19(w30,w18,w11,w5);
assign w31=w18&w11&w6&CIN;
assign C1=w25|w34|w35|w36|w37;
and #(3) and2_20(w34,w17,w26);
```

Information

Module name (8 char. max)
sumator_carry_lc
 Add gate delay info
 Append simul. infomations
 Add labels as comments

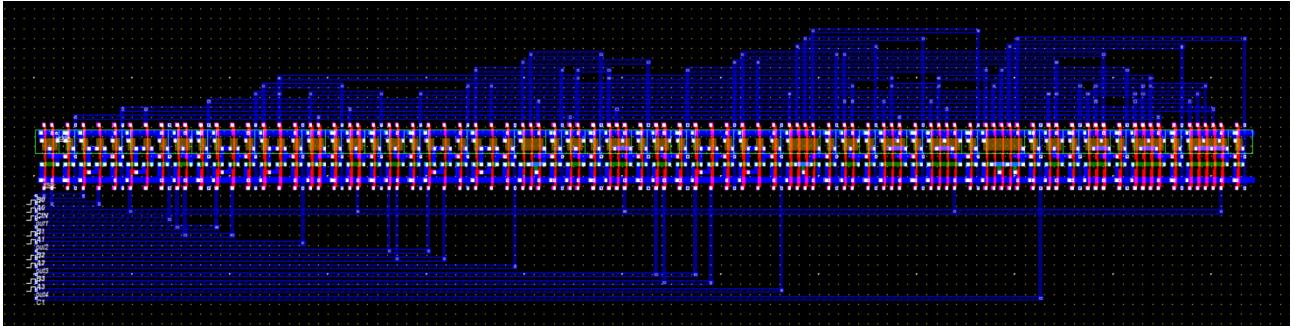
The Verilog file has 62 lines
The design includes 40 symbols
The circuit has 39 nodes

Misc.

Time scale : 1.00
Max clocks: 16

OK

E. Implementarea în Microwind



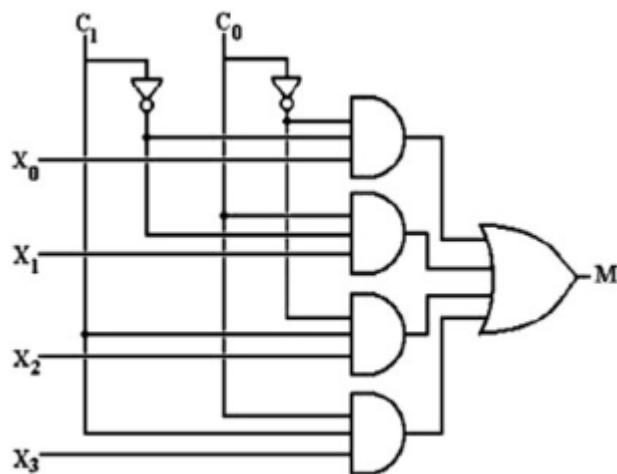
F. Simularea obținută în Microwind

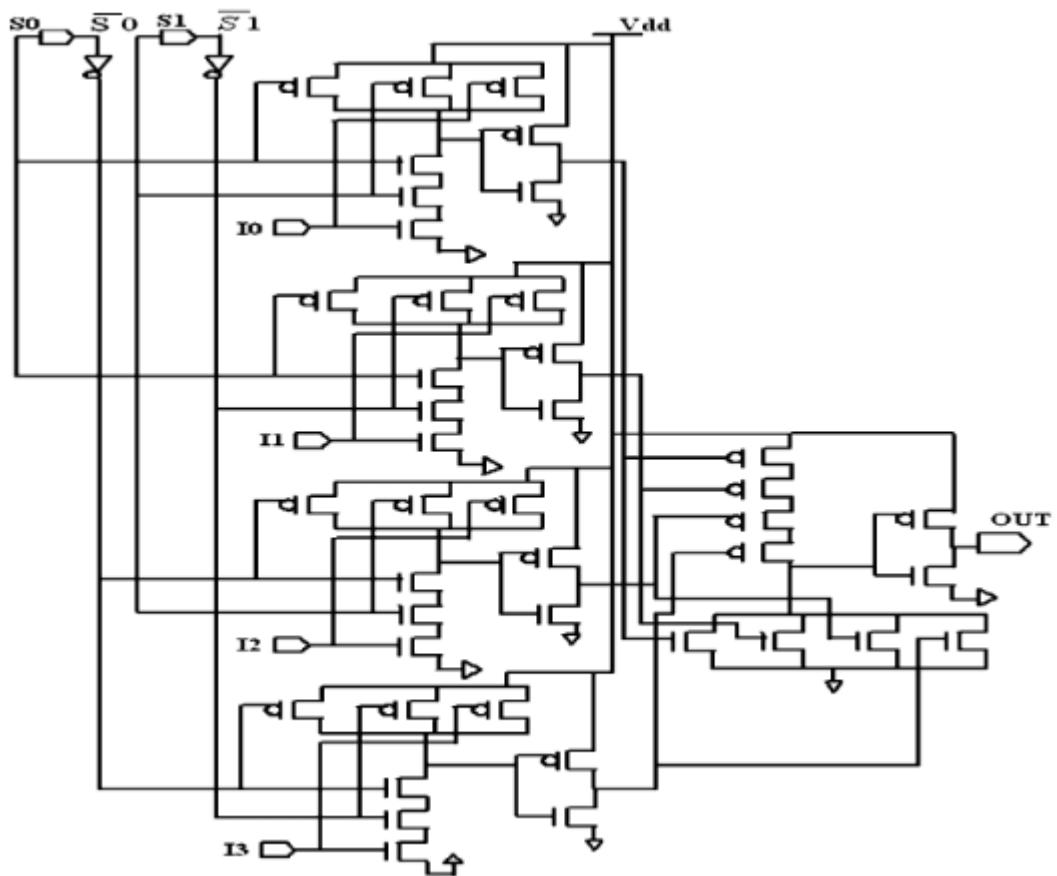


7. MUX 4:1

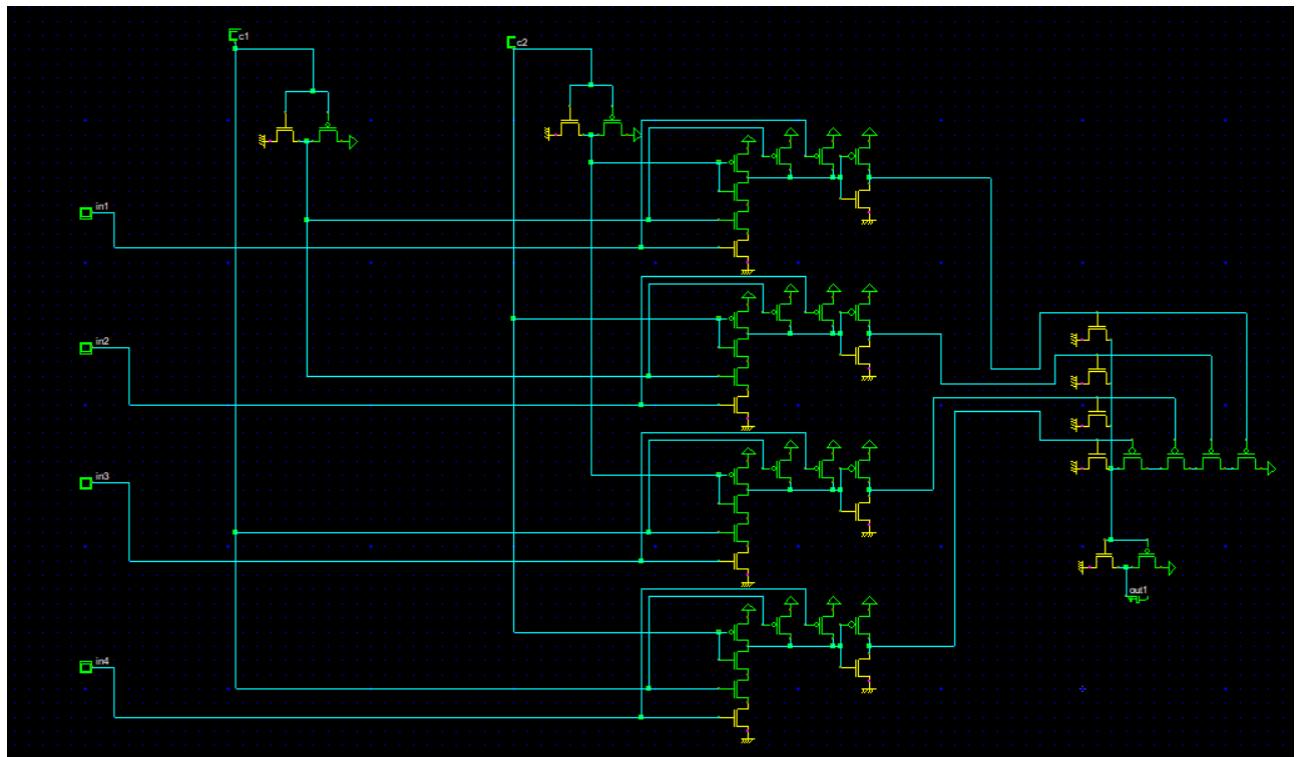
A. Schema circuitului

(sursa:https://www.researchgate.net/publication/257799438_High_performance_low_power_200_Gbs_41_MUX_with_TGL_in_45_nm_technology)

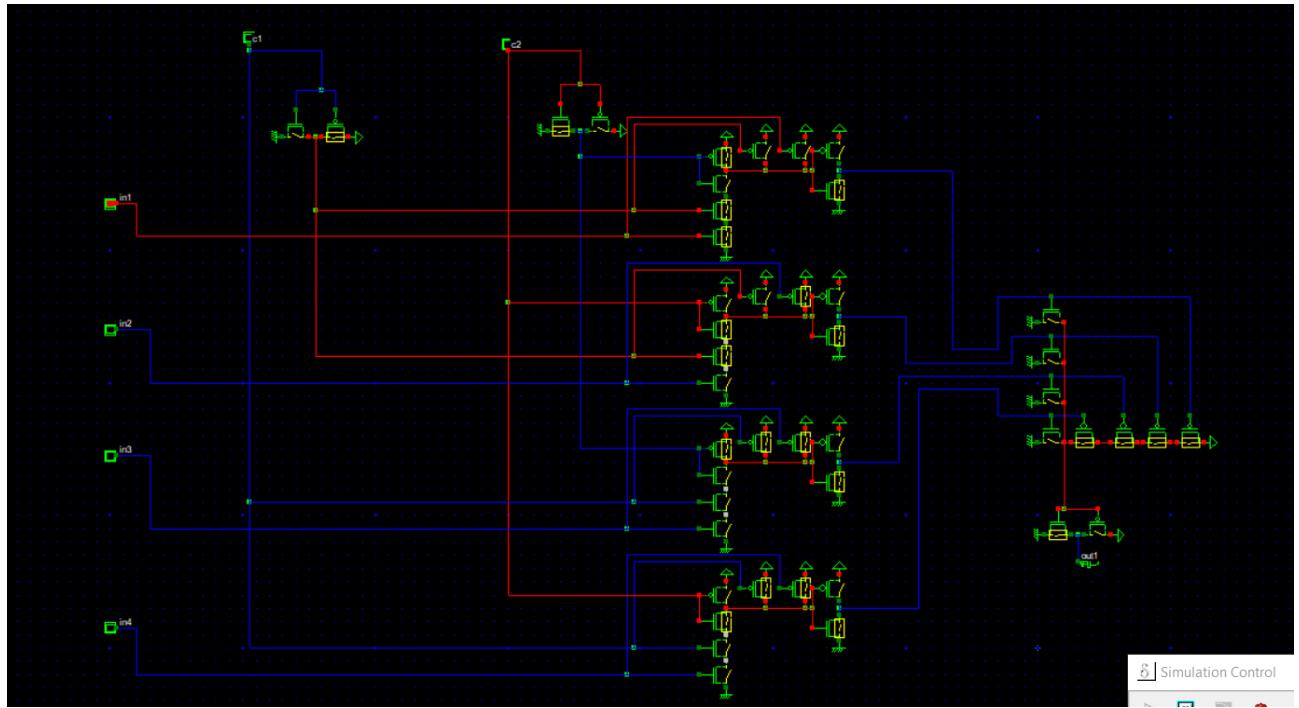




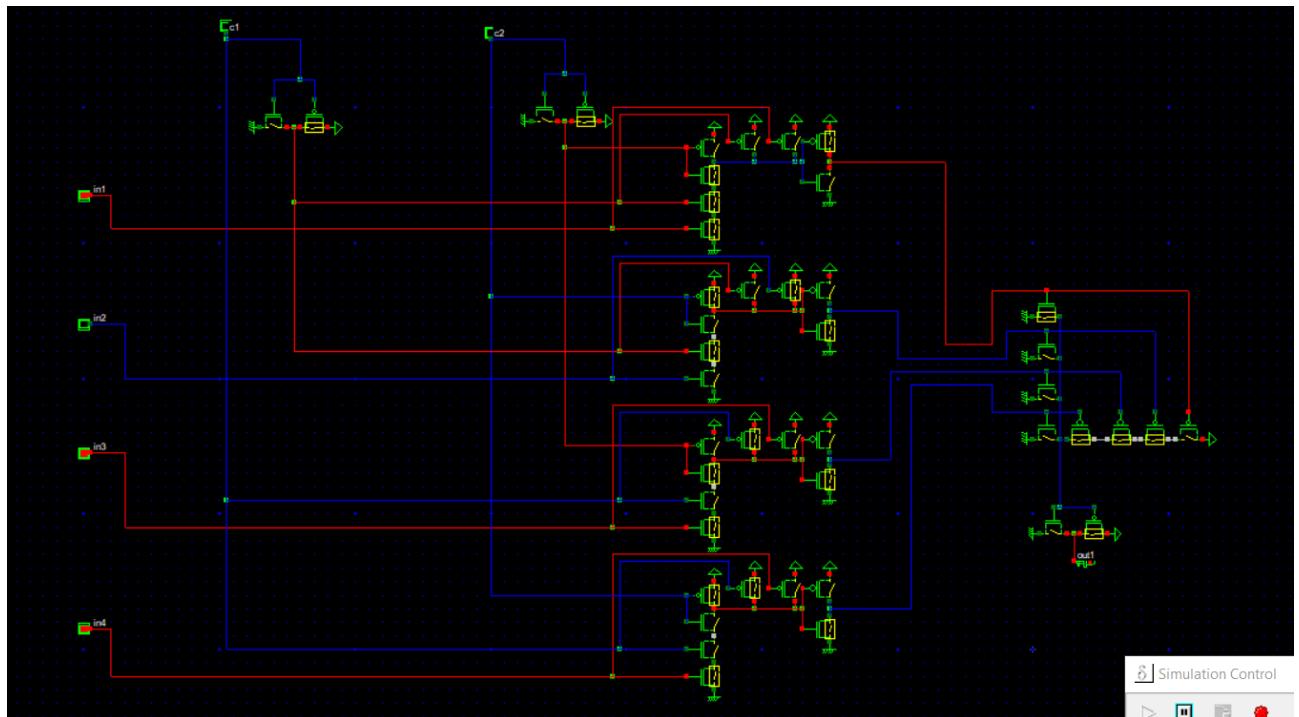
B. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS



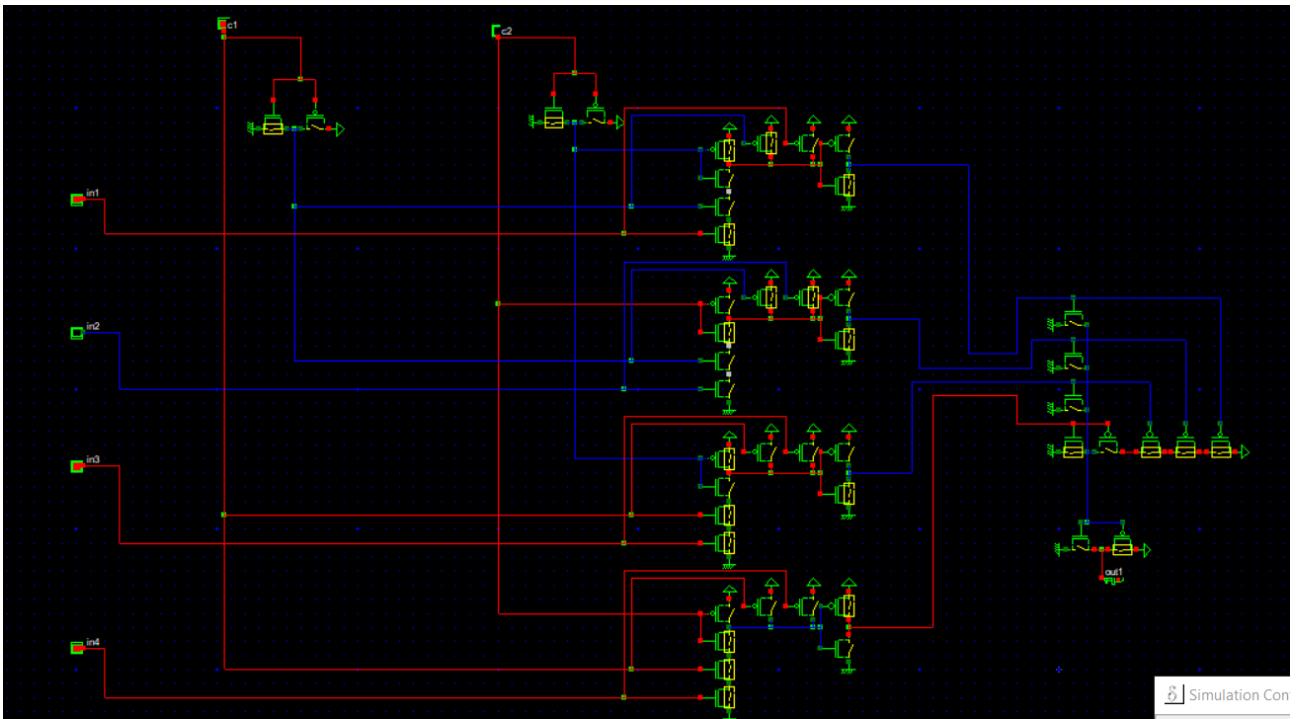
C. Simularea în DSCH



IN1 = 1, IN2, IN3, IN4 = 0, C1 = 0, C2 = 1, OUT1 = 0



IN1 = 1, IN2 = 0, IN3, IN4 = 1, C1, C2=0, OUT1=1



IN1=, IN2=0, IN3, IN4=1, C1=1, C2=1, OUT1=1

D. Codul Verilog obținut

Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path

```
// DSCH 3.5
// 5/11/2023 6:33:46 PM
// D:\VLSI\kituri_VLSI\dsch35_full\examples\MUX4_1.sch

module MUX4_1( inl,in2,in3,in4,c1,c2,out1);
    input inl,in2,in3,in4,c1,c2;
    output out1;
    wire w8,w9,w10,w11,w12,w13,w14,w15;
    wire w16,w17,w18,w19,w20,w21,w22,w23;
    wire w24,w25,w26,w27,w28,w29,;
    pmos #(3) pmos_1(w8,vdd,in1); // 0.5u 0.07u
    pmos #(3) pmos_2(w9,vdd,in4); // 0.5u 0.07u
    pmos #(3) pmos_3(w8,vdd,w10); // 0.5u 0.07u
    pmos #(3) pmos_4(w8,vdd,w11); // 0.5u 0.07u
    pmos #(3) pmos_5(w12,vdd,w11); // 0.5u 0.07u
    nmos #(3) nmos_6(w8,w13,w11); // 0.3u 0.07u
    nmos #(1) nmos_7(w14,vss,in4); // 0.3u 0.07u
    nmos #(3) nmos_8(w9,w15,c2); // 0.3u 0.07u
    pmos #(3) pmos_9(w9,vdd,c2); // 0.5u 0.07u
    nmos #(3) nmos_10(w12,w16,w11); // 0.3u 0.07u
    nmos #(1) nmos_11(w15,w14,c1); // 0.3u 0.07u
    nmos #(2) nmos_12(w17,vss,w9); // 0.3u 0.07u
    pmos #(2) pmos_13(w17,vdd,w9); // 0.5u 0.07u
    pmos #(3) pmos_14(w9,vdd,c1); // 0.5u 0.07u
    pmos #(2) pmos_15(w18,vdd,w8); // 0.5u 0.07u
    nmos #(1) nmos_16(w19,vss,in1); // 0.3u 0.07u
    nmos #(2) nmos_17(w18,vss,w8); // 0.3u 0.07u
    nmos #(1) nmos_18(w13,w19,w10); // 0.3u 0.07u
    pmos #(3) pmos_19(w20,vdd,in2); // 0.5u 0.07u
    nmos #(1) nmos_20(w21,vss,in2); // 0.3u 0.07u
    nmos #(3) nmos_21(w20,w22,c2); // 0.3u 0.07u
    pmos #(3) pmos_22(w20,vdd,c2); // 0.5u 0.07u
    pmos #(3) pmos_23(w12,vdd,in3); // 0.5u 0.07u
    nmos #(1) nmos_24(w22,w21,w10); // 0.3u 0.07u
    nmos #(2) nmos_25(w23,vss,w20); // 0.3u 0.07u
```

Information

Module name (8 char. max)
MUX4_1

Add gate delay info
 Append simul. infomations
 Add labels as comments

The Verilog file has 74 lines
The design includes 88 symbols
The circuit has 30 nodes

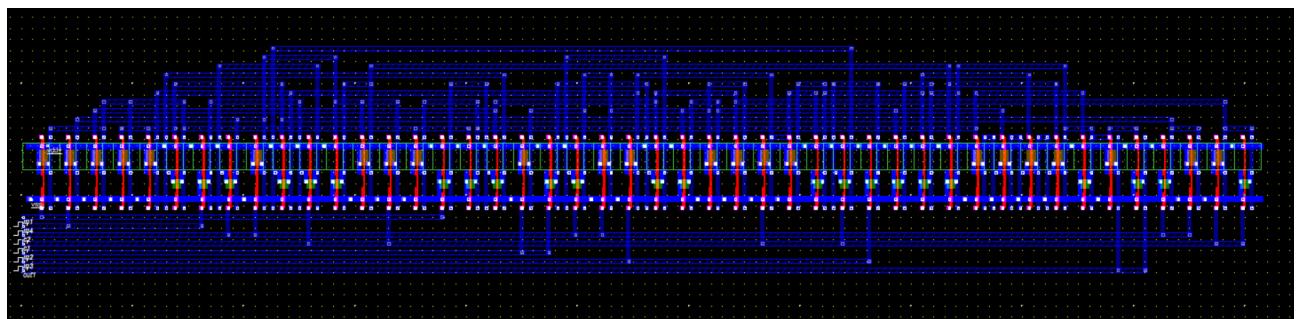
Misc.

Time scale : 1.00
Max clocks: 16

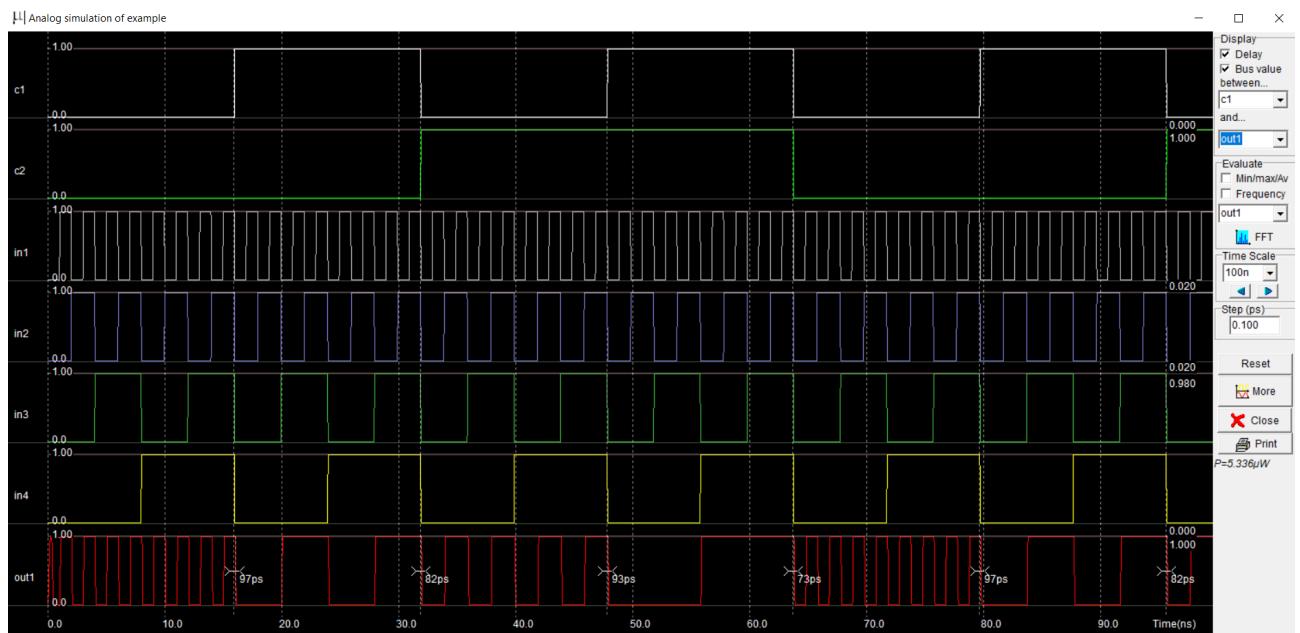
Update Verilog | Extract circuit

OK

E. Implementarea în Microwind

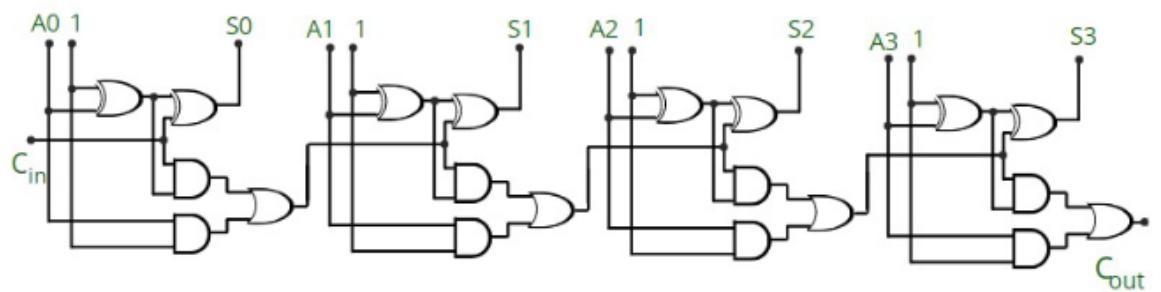
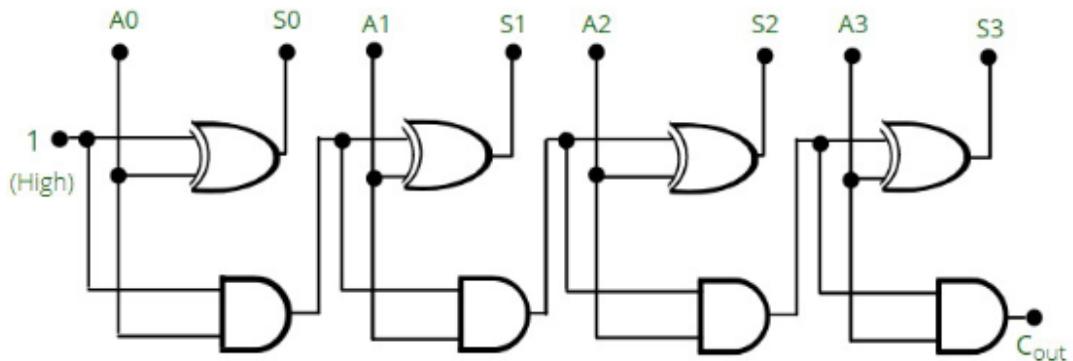


F. Simularea obținută în Microwind



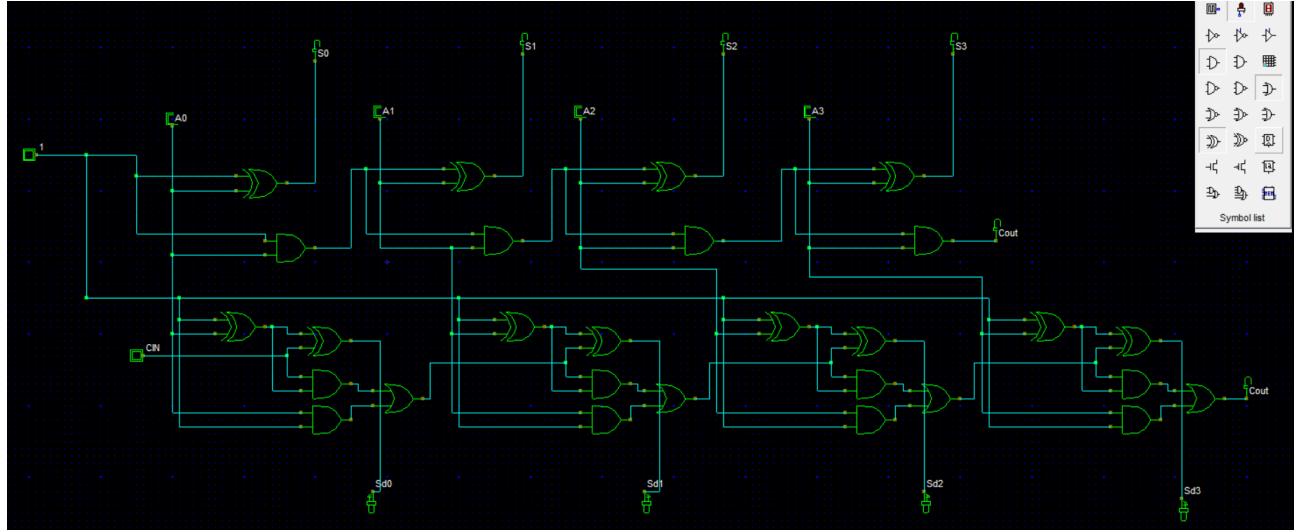
8. Incrementator / decrementator 4 biți

A. Schema circuitului (sursa: <https://www.geeksforgeeks.org/4-bit-binary-incrementer/>)
Incrementator = 4 semisumatoare in serie



Decrementator = 4 sumatoare in serie

B. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS



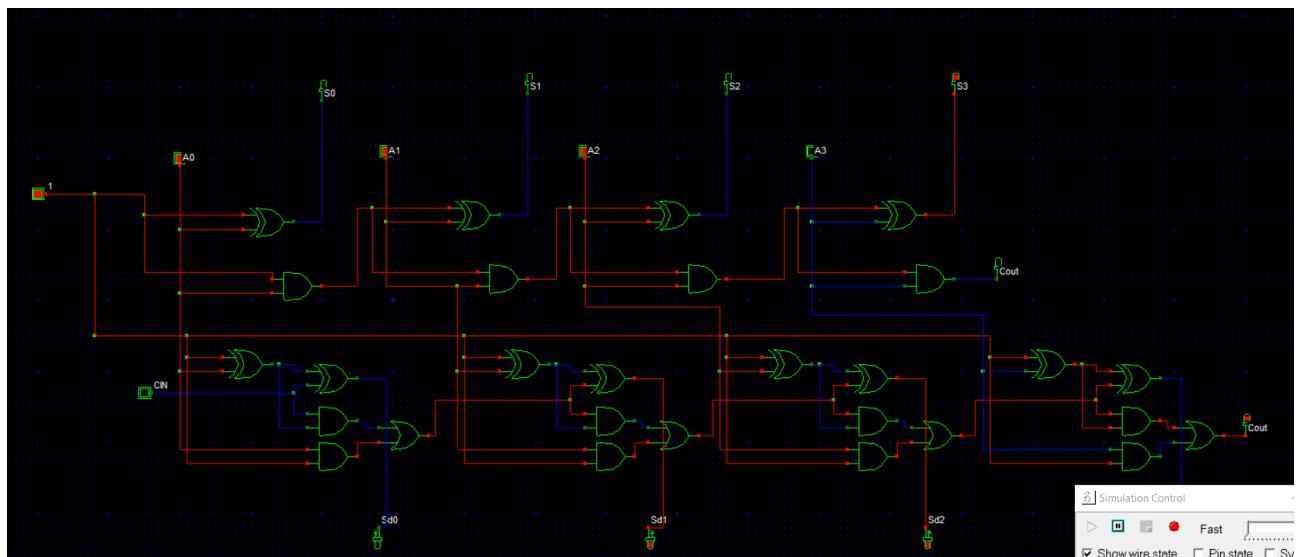
C. Simularea în DSCH

Un numar binar

INPUT: A3A2A1A0

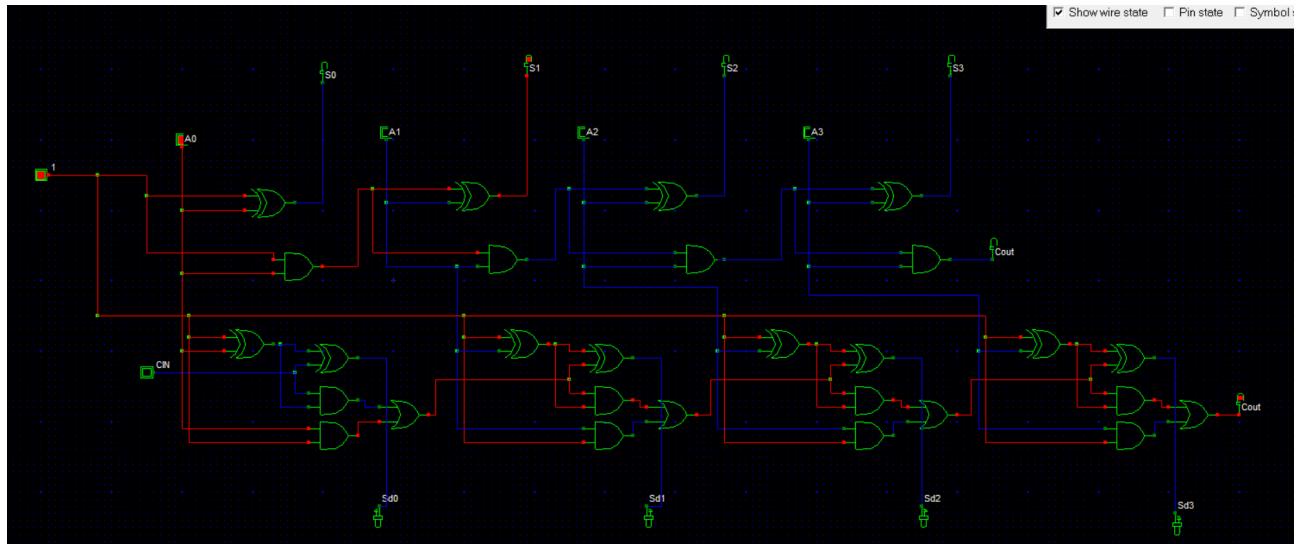
Output Incrementator: S3S2S1S0

Output Decrementator: Sd3Sd2Sd1Sd0



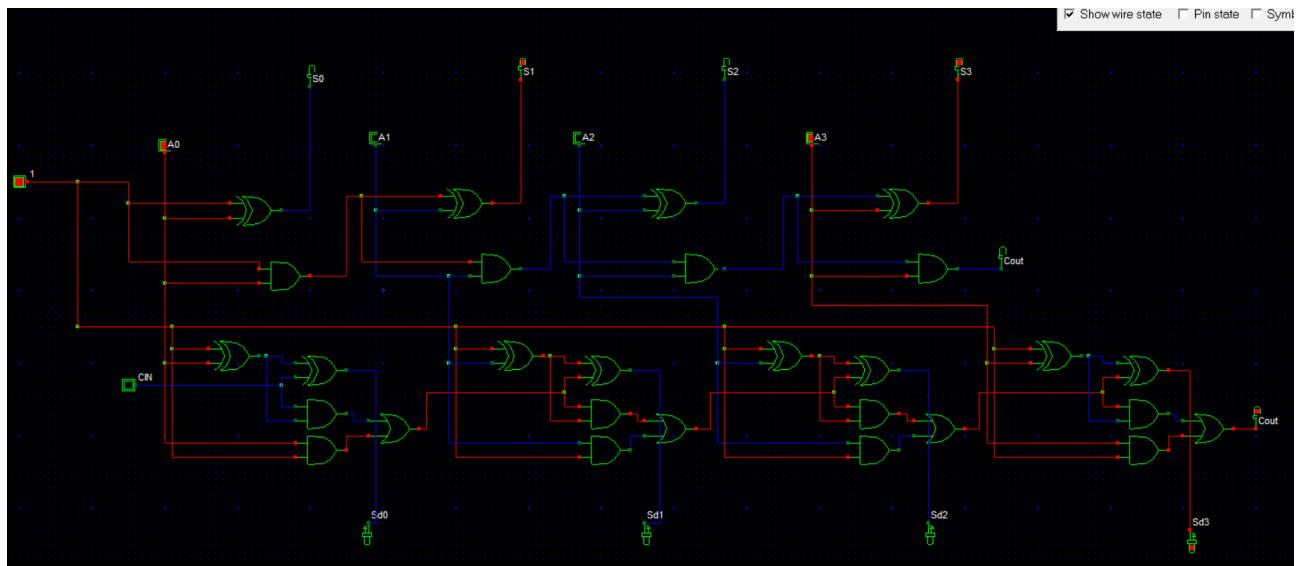
$$0111 \text{ (A3A2A1A0)} + 1 = 1000 \text{ (S3S2S1S0)}$$

$$0111 \text{ (A3A2A1A0)} - 1 = 0110 \text{ (Sd3Sd2Sd1Sd0)}$$



$$0001 \text{ (A}_3\text{A}_2\text{A}_1\text{A}_0\text{)} + 1 = 0010 \text{ (S}_3\text{S}_2\text{S}_1\text{S}_0\text{)}$$

$$0001 \text{ (A}_3\text{A}_2\text{A}_1\text{A}_0\text{)} - 1 = 0000 \text{ (S}_d3\text{S}_d2\text{S}_d1\text{S}_d0\text{)}$$



$$1001 \text{ (A}_3\text{A}_2\text{A}_1\text{A}_0\text{)} + 1 = 1010 \text{ (S}_3\text{S}_2\text{S}_1\text{S}_0\text{)}$$

$$1001 \text{ (A}_3\text{A}_2\text{A}_1\text{A}_0\text{)} - 1 = 1000 \text{ (S}_d3\text{S}_d2\text{S}_d1\text{S}_d0\text{)}$$

D. Codul Verilog obținut

Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path |

```

module incrementer_and_decrementer( A3,c11,A0,A1,A2,CIN,S0,S3,
    S1,S2,Cout,Cout,Sd0,Sd1,Sd2,Sd3);
    input A3,c11,A0,A1,A2,CIN;
    output S0,S3,S1,S2,Cout,Cout,Sd0,Sd1;
    output Sd2,Sd3;
    wire w4,w9,w12,w15,w16,w17,w18,w19;
    wire w20,w21,w22,w24,w25,w27,w28,w30;
    wire w32,w33,w34,w35,w36,w37,w38,w40;
    wire w42,w43;
    and #(3) and2_1(Cout,A3,w4);
    xor #(3) xor2_2(S3,w4,A3);
    and #(4) and2_3(w9,A0,c11);
    xor #(3) xor2_4(S0,c11,A0);
    xor #(3) xor2_5(S1,w9,A1);
    and #(4) and2_6(w12,A1,w9);
    and #(4) and2_7(w4,A2,w12);
    xor #(3) xor2_8(S2,w12,A2);
    or #(3) or2_9(w17,w15,w16);
    and #(3) and2_10(w16,w18,w19);
    and #(3) and2_11(w15,w20,w21);
    xor #(3) xor2_12(w22,w20,w21);
    xor #(4) xor2_13(w20,w18,w19);
    or #(3) or2_14(Cout,w24,w25);
    and #(3) and2_15(w25,c11,A3);
    and #(3) and2_16(w24,w27,w28);
    xor #(3) xor2_17(Sd3,w27,w28);
    xor #(4) xor2_18(w27,c11,A3);
    xor #(4) xor2_19(w30,c11,A0);
    xor #(3) xor2_20(Sd0,w30,CIN);
    and #(3) and2_21(w32,w30,CIN);
    and #(3) and2_22(w33,c11,A0);
    or #(4) or2_23(w34,w32,w33);
    or #(4) or2_24(w37,w35,w36);
    and #(3) and2_25(w36,c11,A1);

```

Information

Module name (8 char. max)
incrementer_and

Add gate delay info
 Append simul. informations
 Add labels as comments

The Verilog file has 64 lines
The design includes 49 symbols
The circuit has 43 nodes

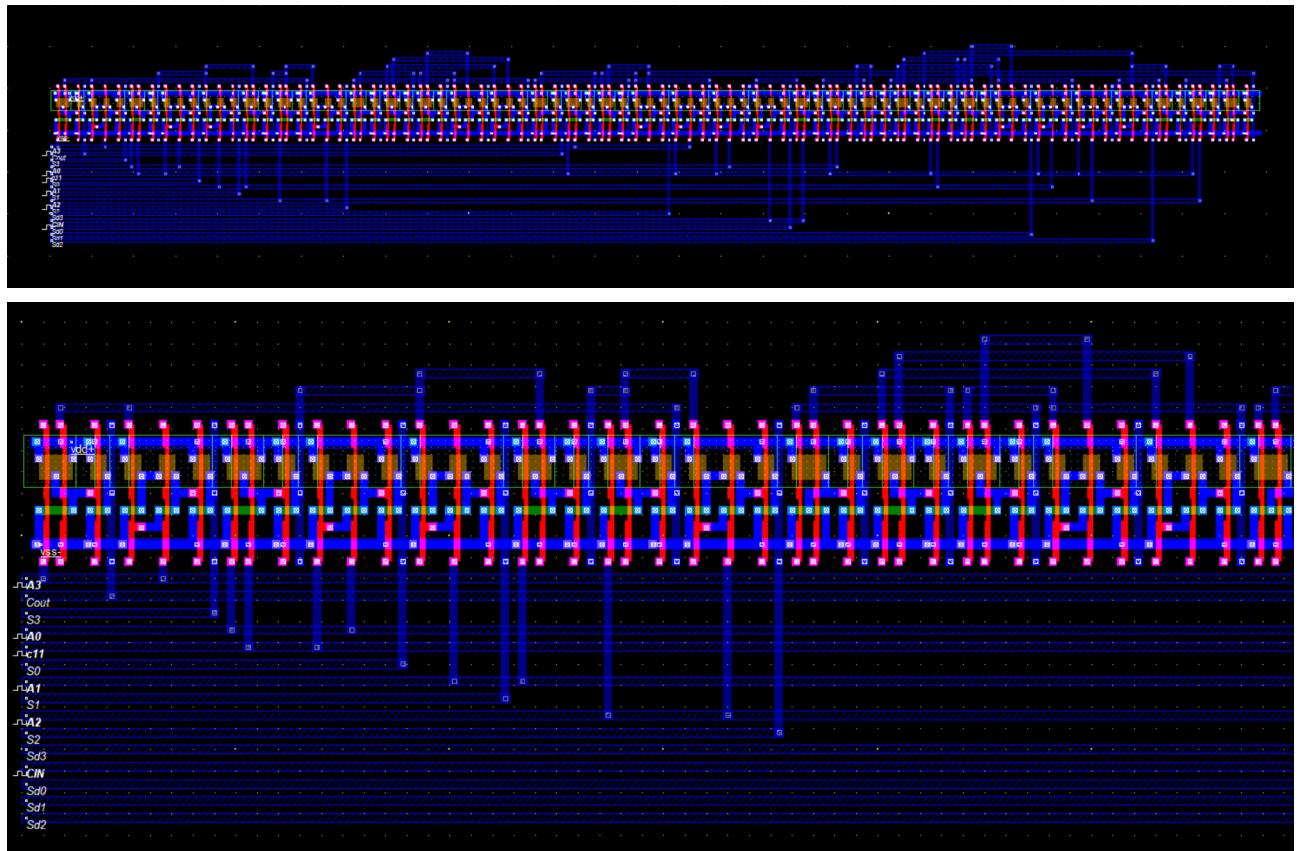
Misc.

Time scale : 1.00
Max clocks: 16

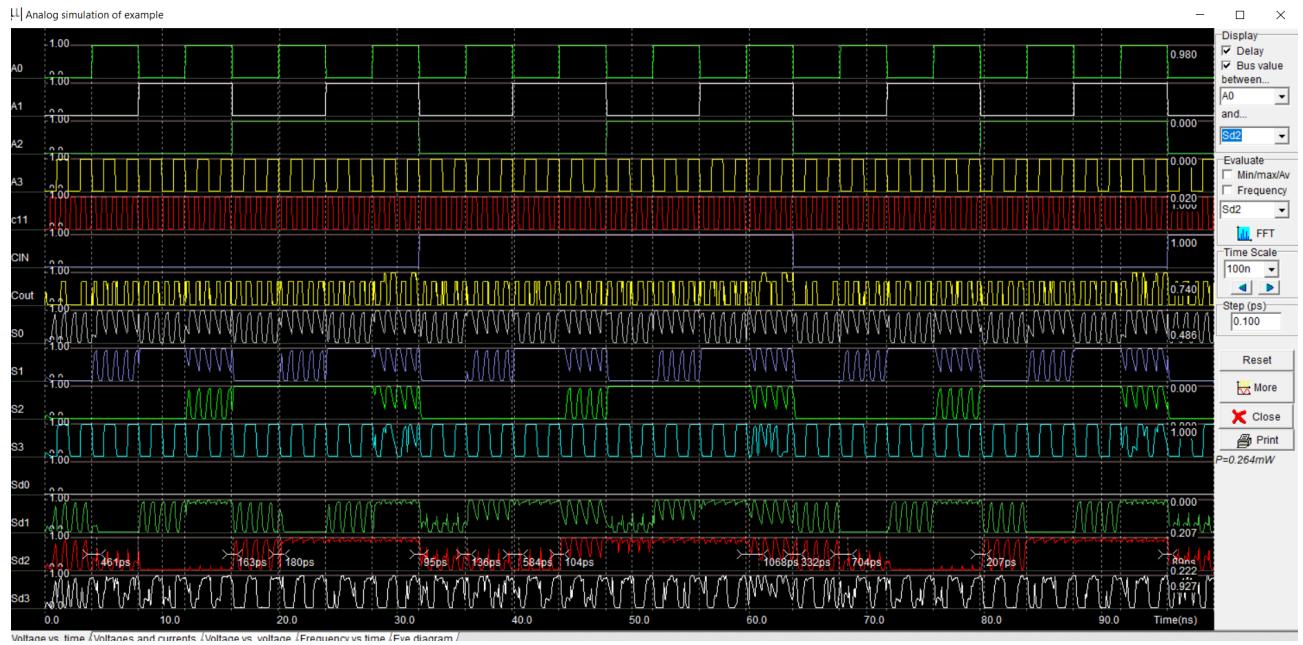
Update Verilog Extract circuit

OK

E. Implementarea în Microwind



F. Simularea obținută în Microwind



9. Registru deplasare stânga/dreapta 4 biți

A. Schema circuitului (sursa:<http://ep/etc.tuiasi.ro/files/CID/registre.pdf>,
<https://www.ques10.com/p/6634/what-is-shift-register-explain-4-bit-bi-directional/>)

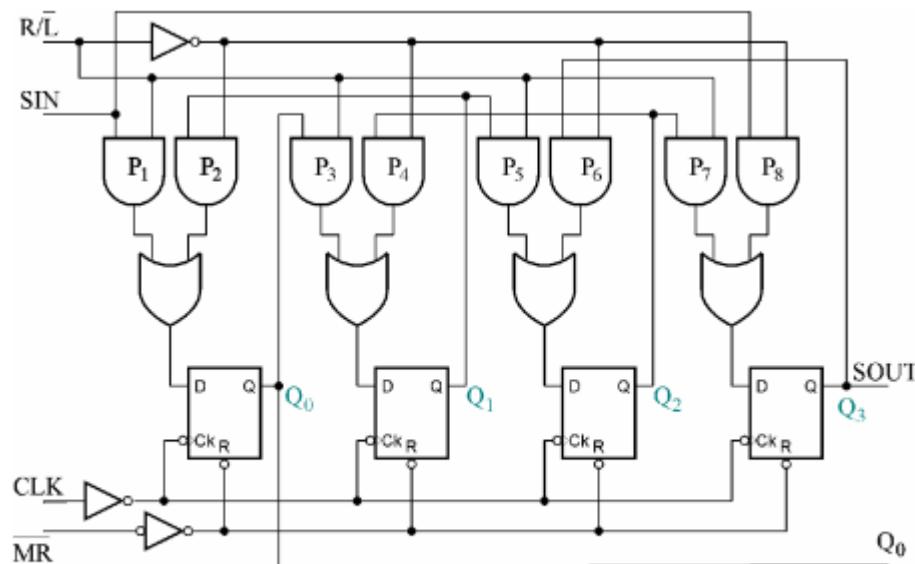
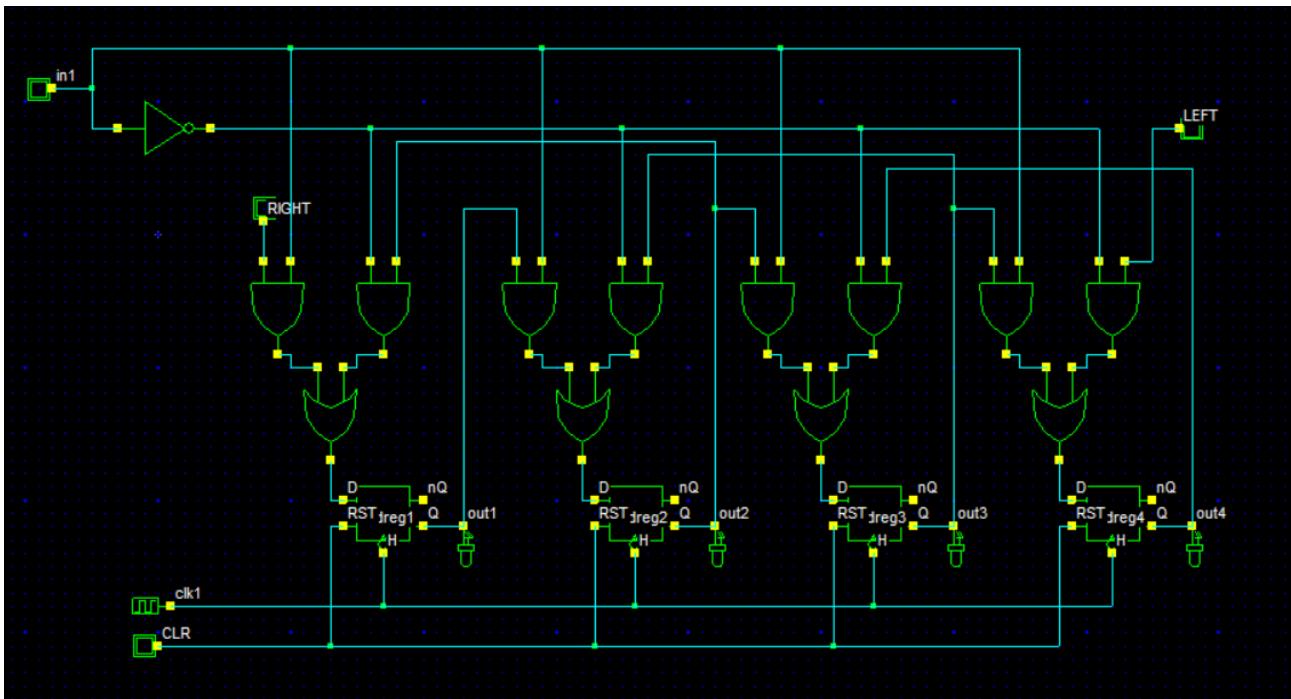


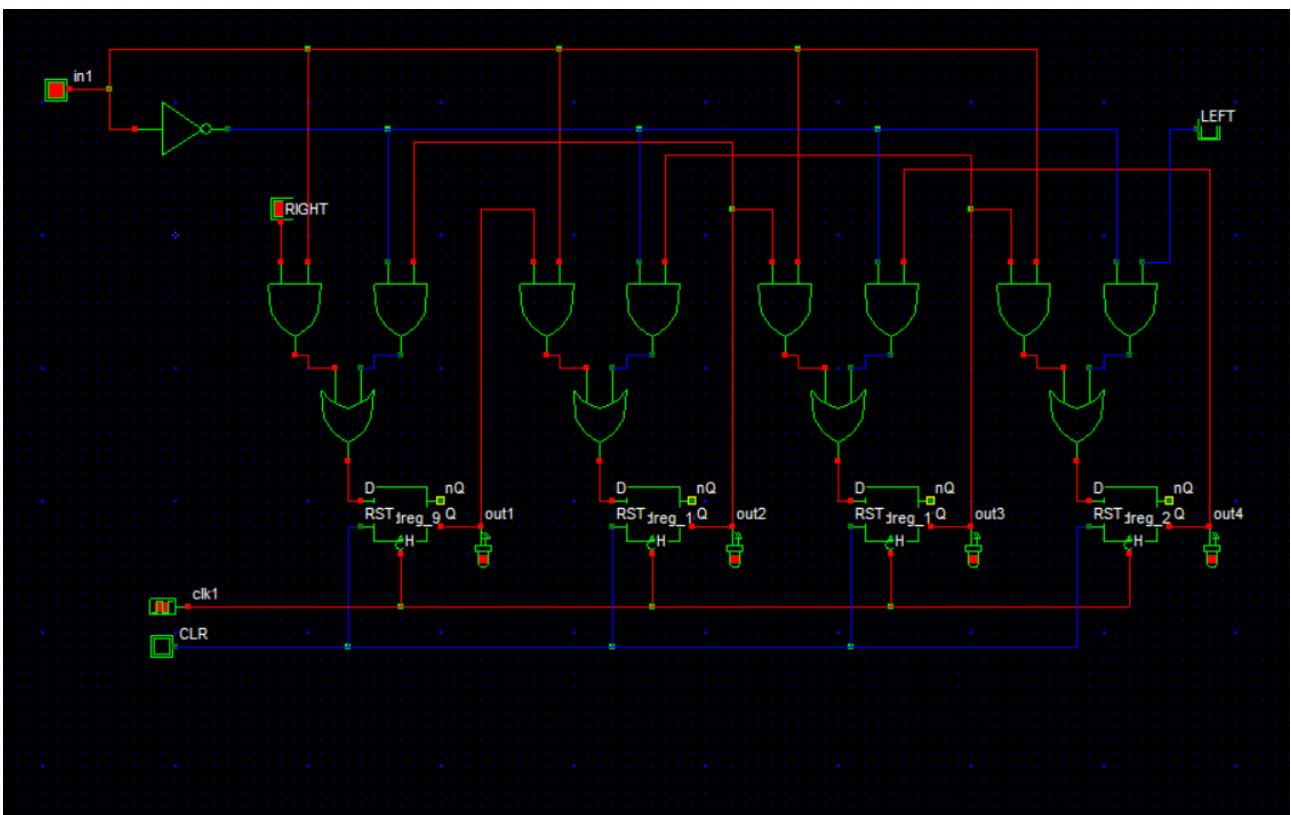
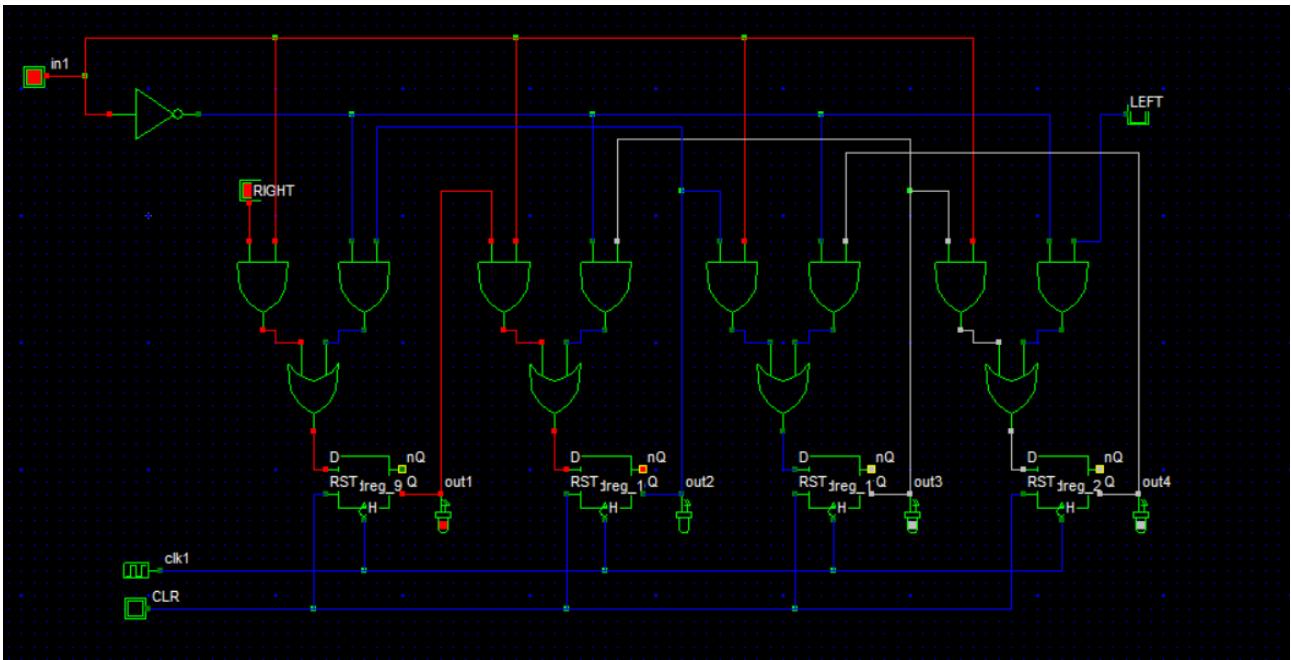
Figura 3.7. Registrul SISO de 4 biti bidirectional.

B. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS

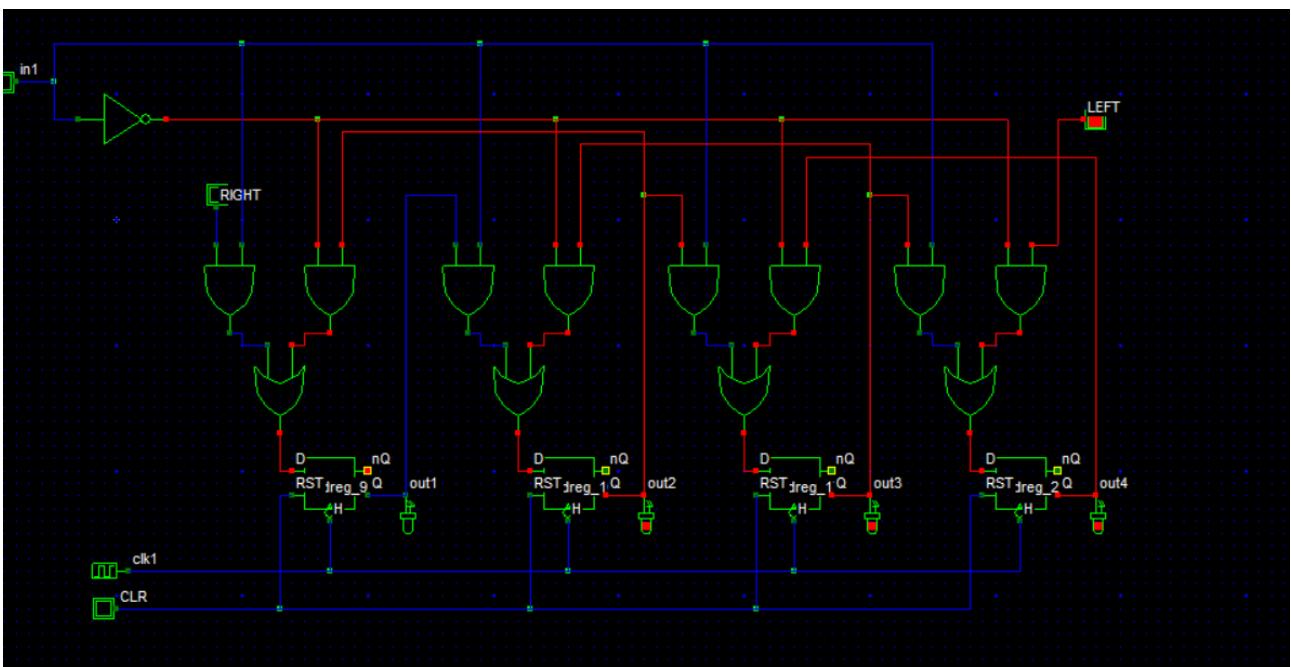
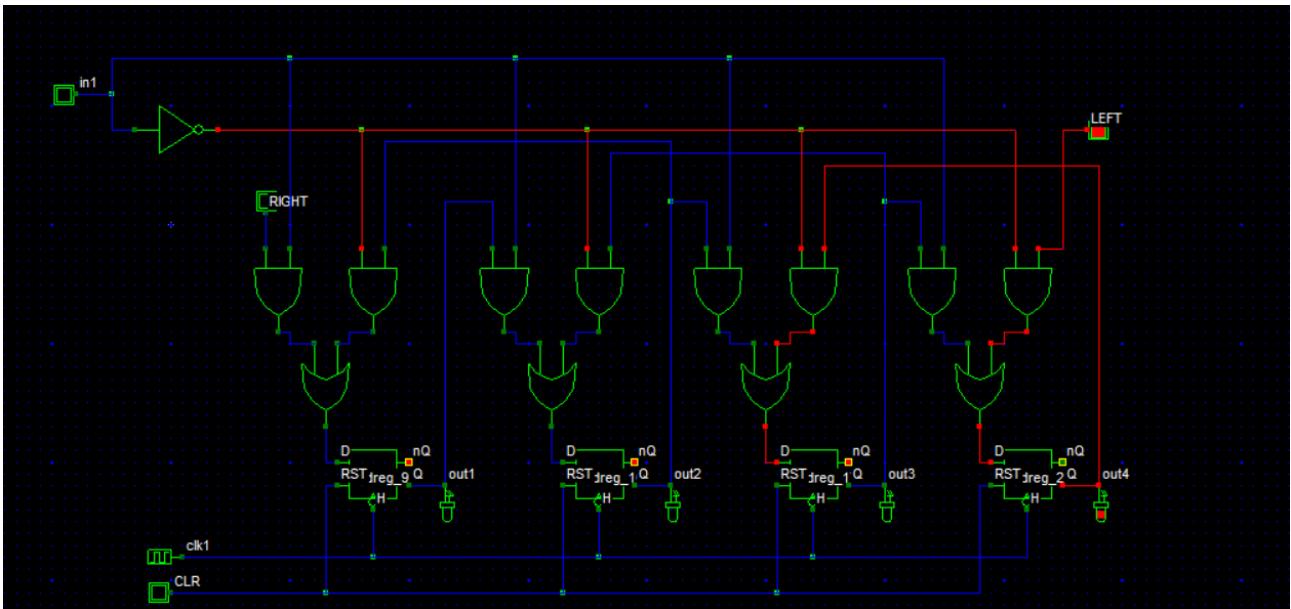


C. Simularea în DSCH

RIGHT



LEFT



D. Codul Verilog obținut

Verilog, Hierarchy and Netlist | Critical path

```

// DSCH 3.5
// 5/11/2023 11:32:50 PM
// D:\VLSI\kituri_VLSI\dsch35_full\examples\registrul_deplasare_stanga_dreapta.v

module registrul_deplasare_stanga_dreapta( in1,RIGHT,CLR,clk1,LEFT;
output out4,out1,out2,out3;
wire w3,w7,w9,w10,w11,w14,w16,w17;
wire w19,w20,w21,w22,w23,w24,w25,w26;
wire w27;
not #(3) inv_1(w3,in1);
dreg #(4) dreg_2(out4,w9,w7,CLR,clk1);
or #(3) or2_3(w7,w10,w11);
and #(3) and2_4(w10,w3,LEFT);
and #(3) and2_5(w11,out3,in1);
and #(3) and2_6(w14,RIGHT,in1);
and #(3) and2_7(w16,w3,out2);
or #(3) or2_8(w17,w16,w14);
dreg #(4) dreg_9(out1,w19,w17,CLR,clk1);
dreg #(4) dreg_10(out2,w21,w20,CLR,clk1);
or #(3) or2_11(w20,w22,w23);
and #(3) and2_12(w22,w3,out3);
and #(3) and2_13(w23,out1,in1);
and #(3) and2_14(w24,out2,in1);
and #(3) and2_15(w25,w3,out4);
or #(3) or2_16(w26,w25,w24);
dreg #(4) dreg_17(out3,w27,w26,CLR,clk1);
endmodule

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

```

Information

Module name (8 char. max)
registrul_deplasare

Add gate delay info
 Append simul. infomations
 Add labels as comments

The Verilog file has 44 lines
The design includes 26 symbols
The circuit has 27 nodes

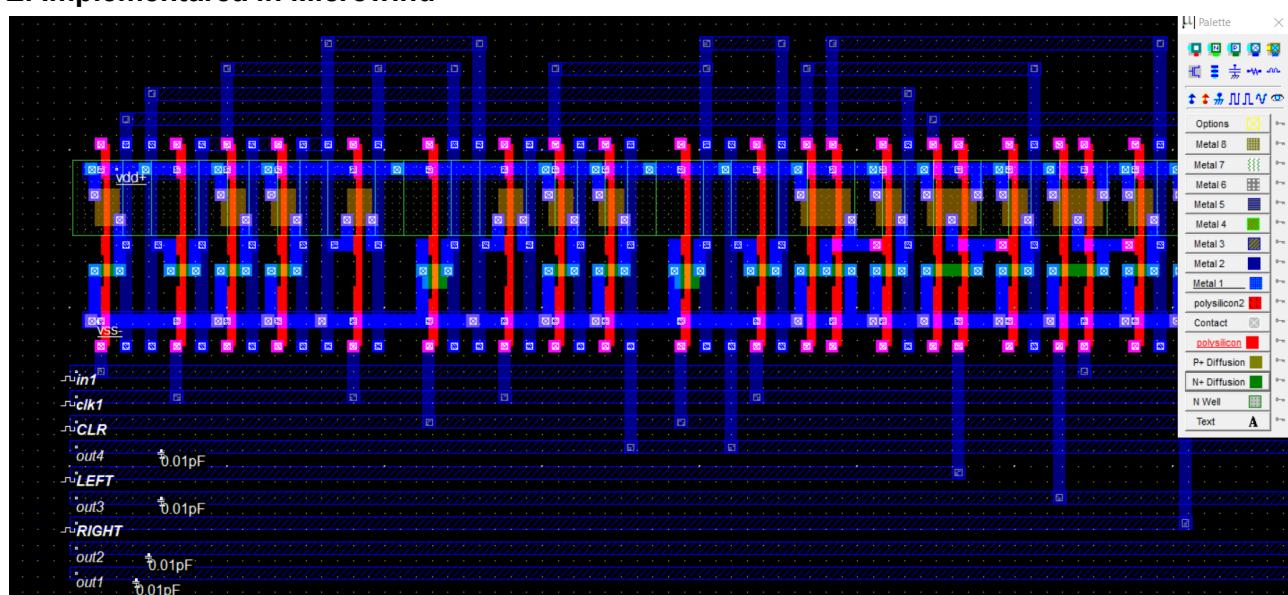
Misc.

Time scale : 1.00
Max clocks: 16

Update Verilog Extract circuit

OK

E. Implementarea în Microwind



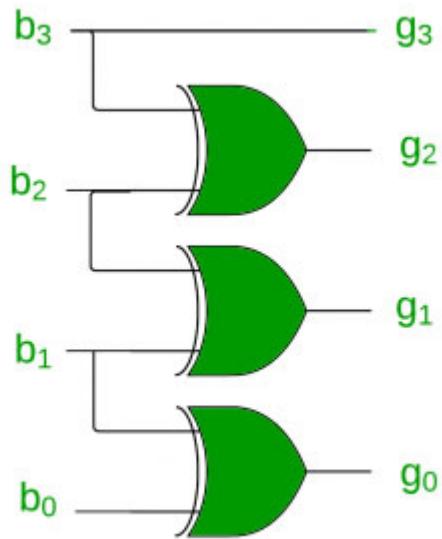
F. Simularea obținută în Microwind



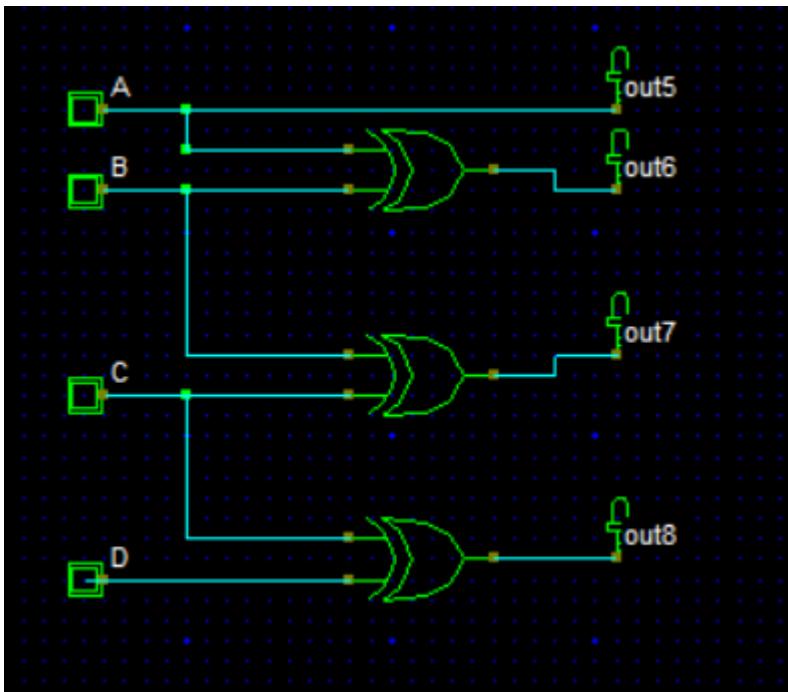
10. Convertor cod Gray 4 biți

A. Schema circuitului

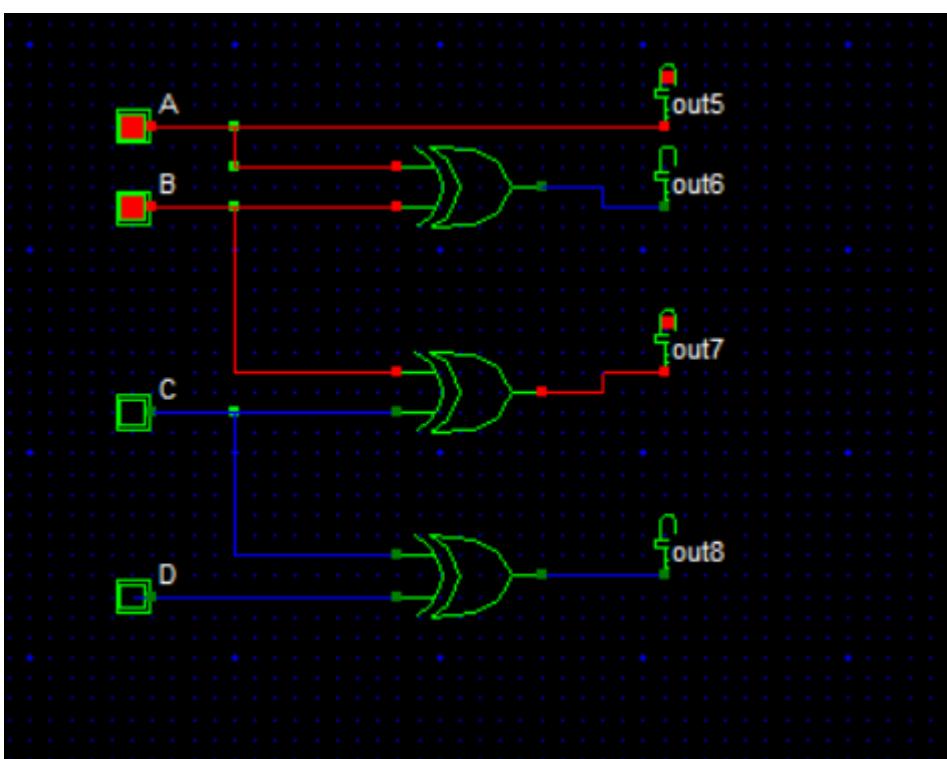
(sursa: <https://www.geeksforgeeks.org/code-converters-binary-to-from-gray-code/>)



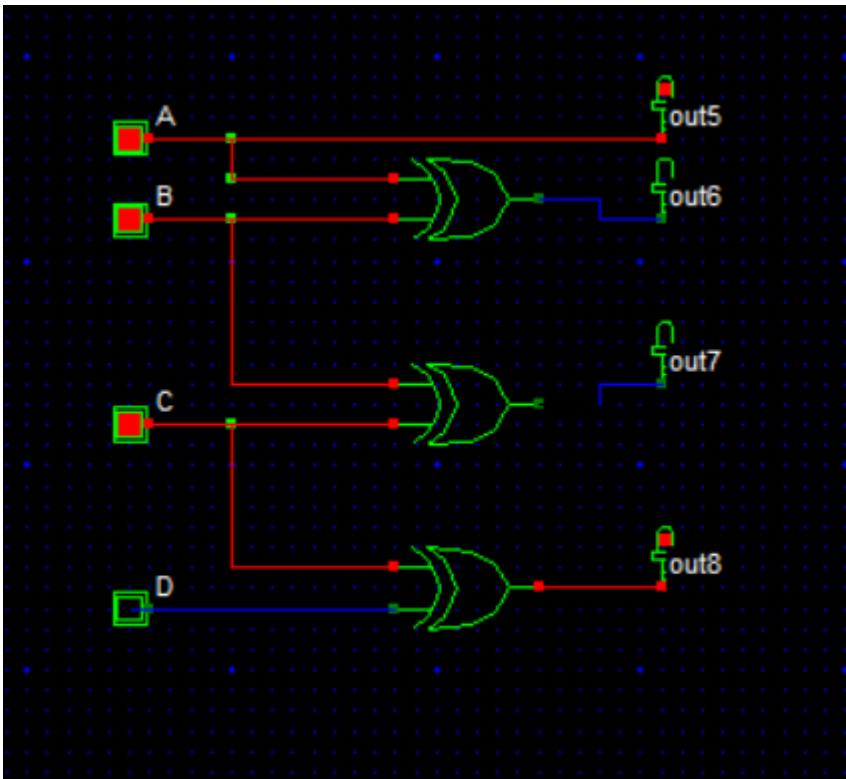
B. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS



C. Simularea în DSCH



1100 -> Gray (1010)



1110 -> Gray(1001)

D. Codul Verilog obținut

Verilog Hierarchy Netlist Critical path

```
// DSCH 3.5
// 5/11/2023 10:40:37 PM
// D:\VLSI\kituri_VLSI\dsch35_full\examples\binary_to_gray.v

module binary_to_gray( B0,B2,B1,B3,G0,G1,G2,G3);
    input B0,B2,B1,B3;
    output G0,G1,G2,G3;
    wire ;
    xor #(3) xor2_1(G0,B1,B0);
    xor #(3) xor2_2(G1,B2,B1);
    xor #(3) xor2_3(G2,B3,B2);
endmodule

// Simulation parameters in Verilog Format
always
#200 B0=~B0;
#400 B2=~B2;
#800 B1=~B1;
#1600 B3=~B3;

// Simulation parameters
// B0 CLK 1 1
// B2 CLK 2 2
// B1 CLK 4 4
// B3 CLK 8 8
```

Information

Module name (8 char. max)
binary_to_gray

Add gate delay info
 Append simul. infomations
 Add labels as comments

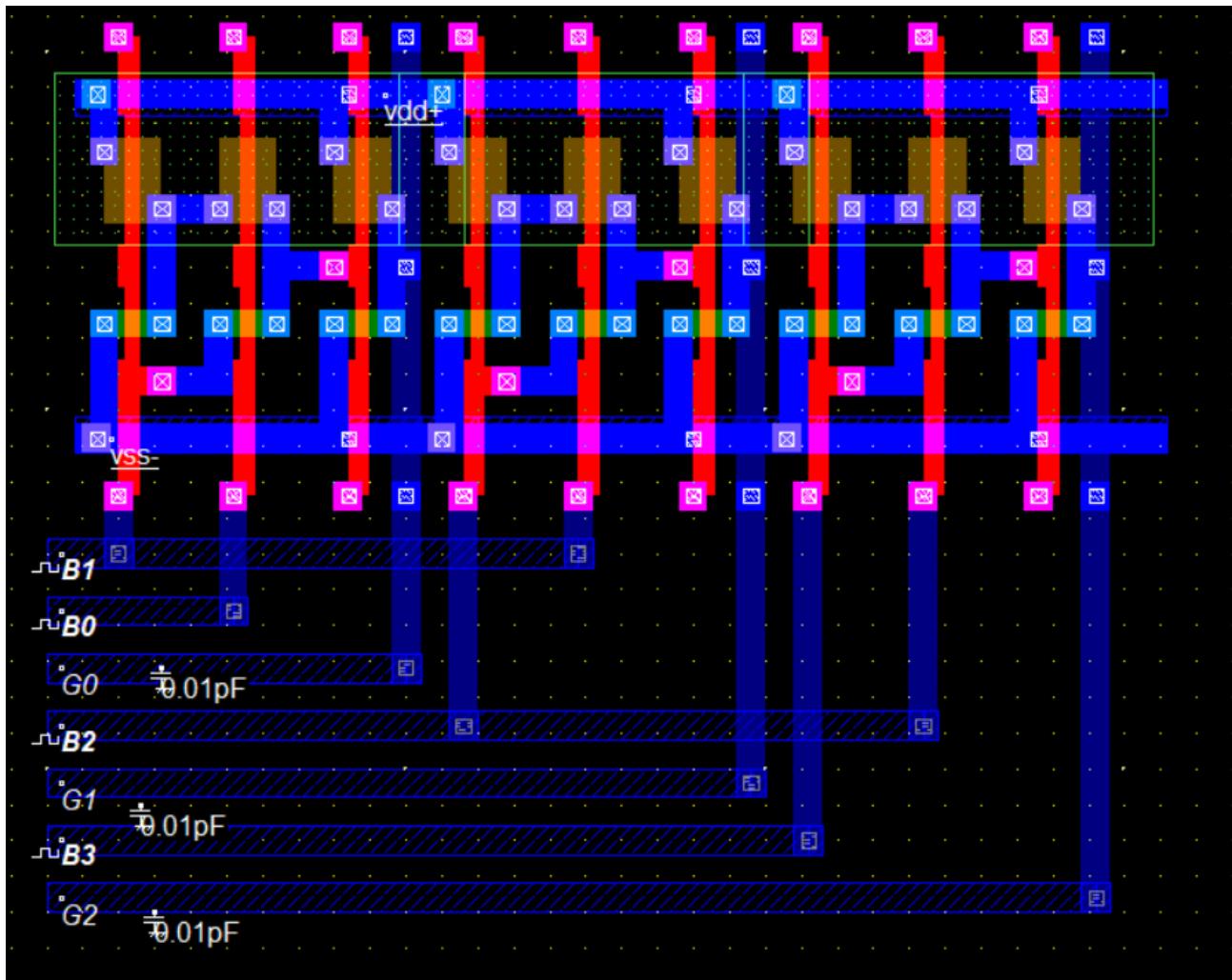
The Verilog file has 25 lines
The design includes 11 symbols
The circuit has 8 nodes

Misc.

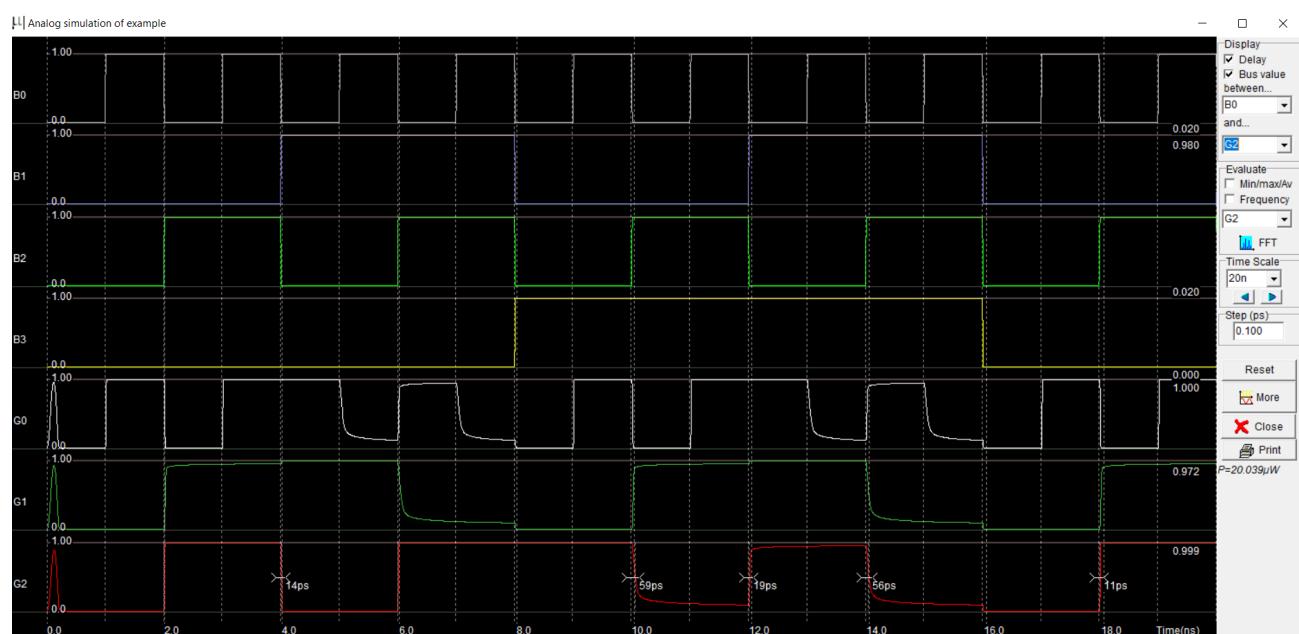
Time scale : 1.00
Max clocks: 16

Update Verilog Extract circuit

E. Implementarea în Microwind



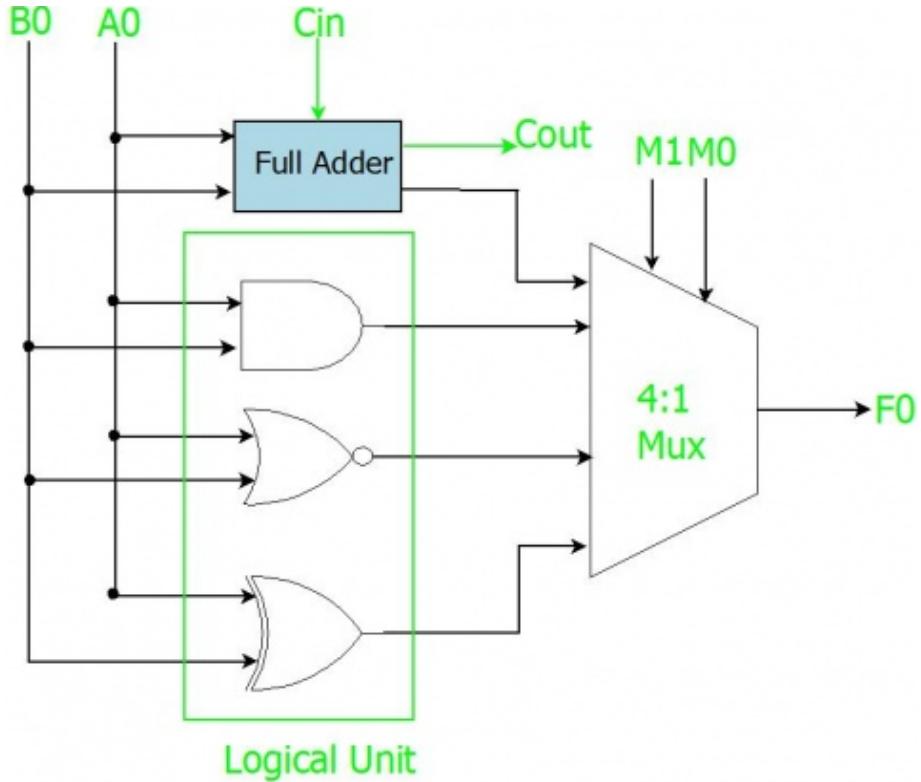
F. Simularea obținută în Microwind



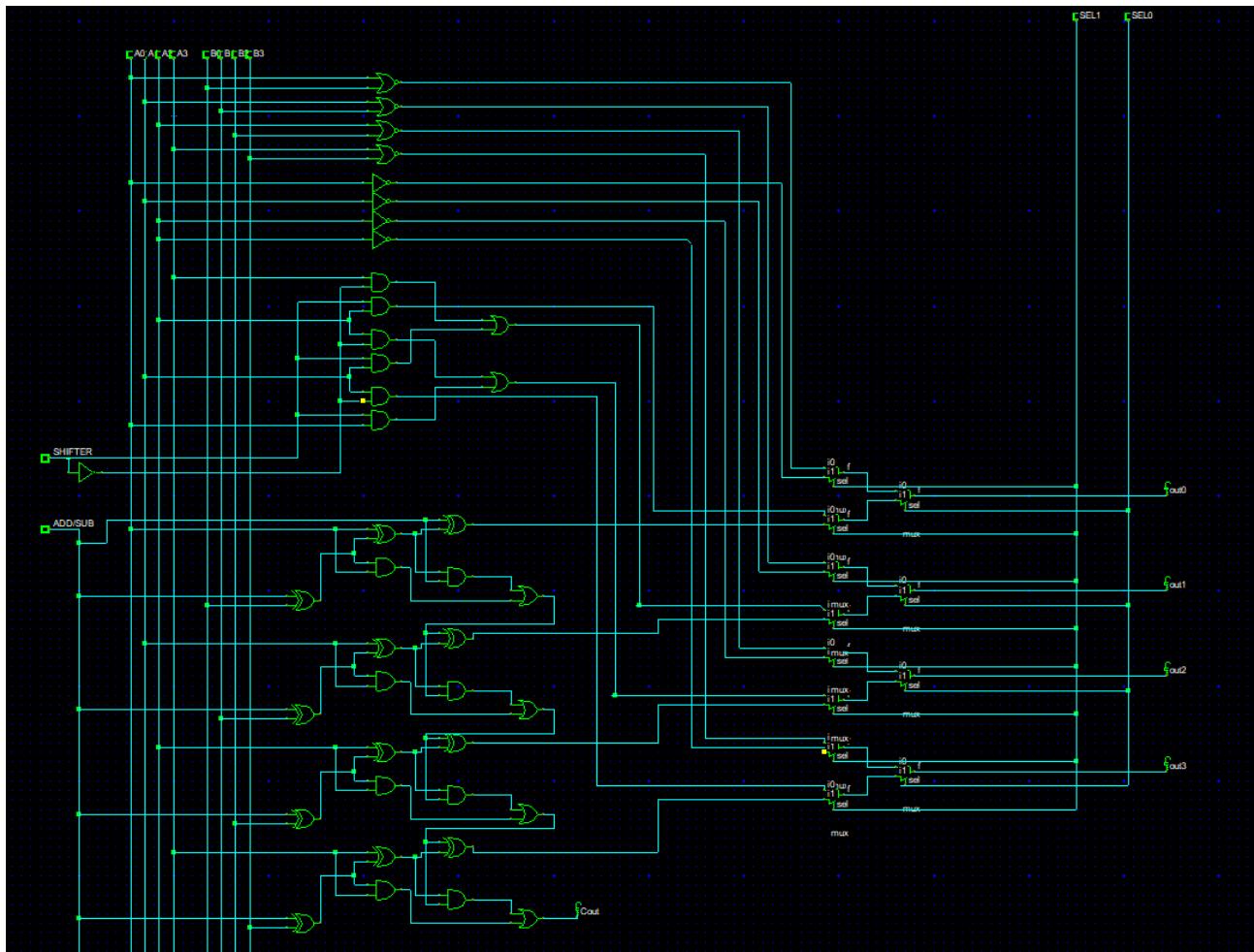
11. ALU pe 4 biți (not, +, -, nor, shr, shl)

A. Schema circuitului

(sursa:https://exploreembedded.com/wiki/ALU_in_Detail)

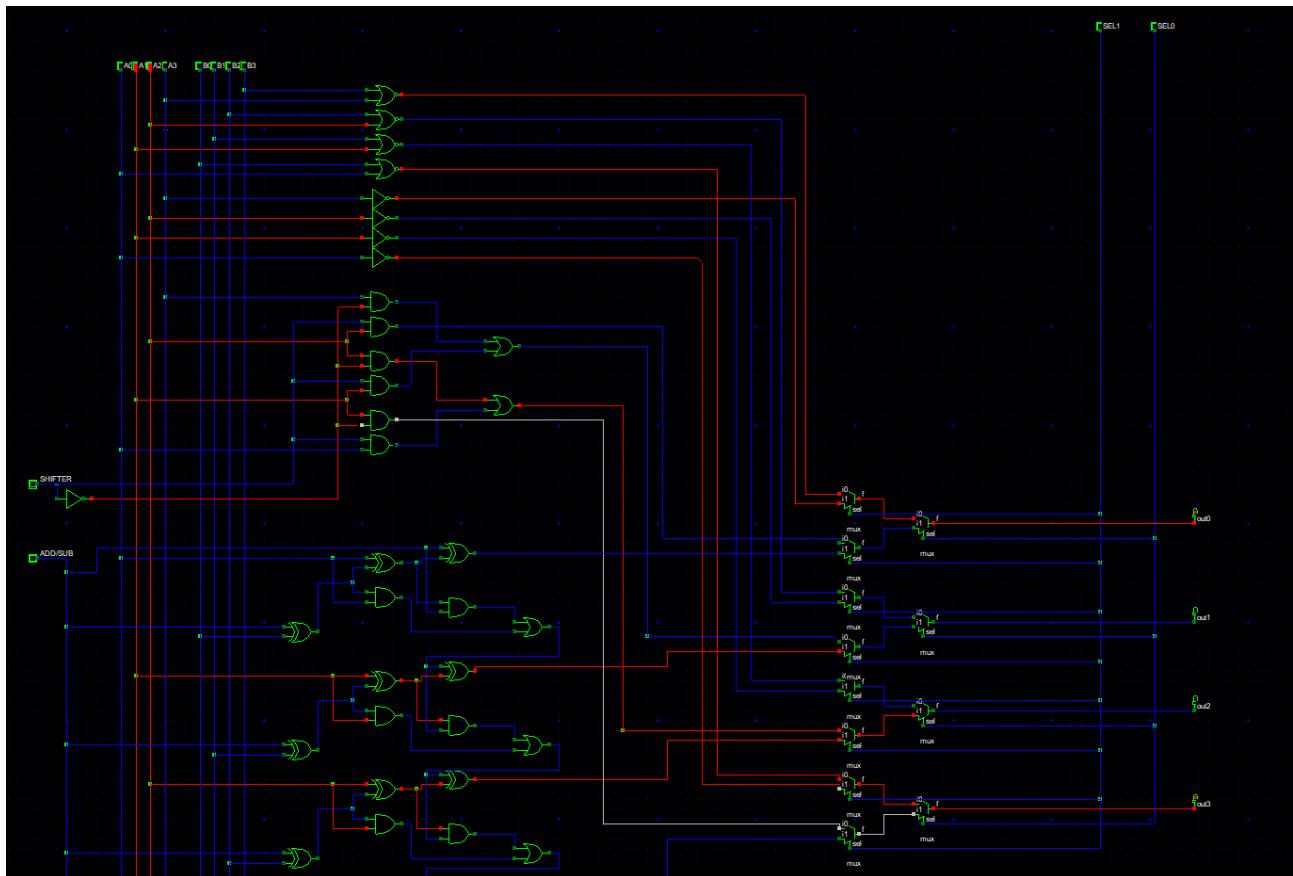


B. Implementarea în DSCH la nivel de tranzistor nMOS și pMOS

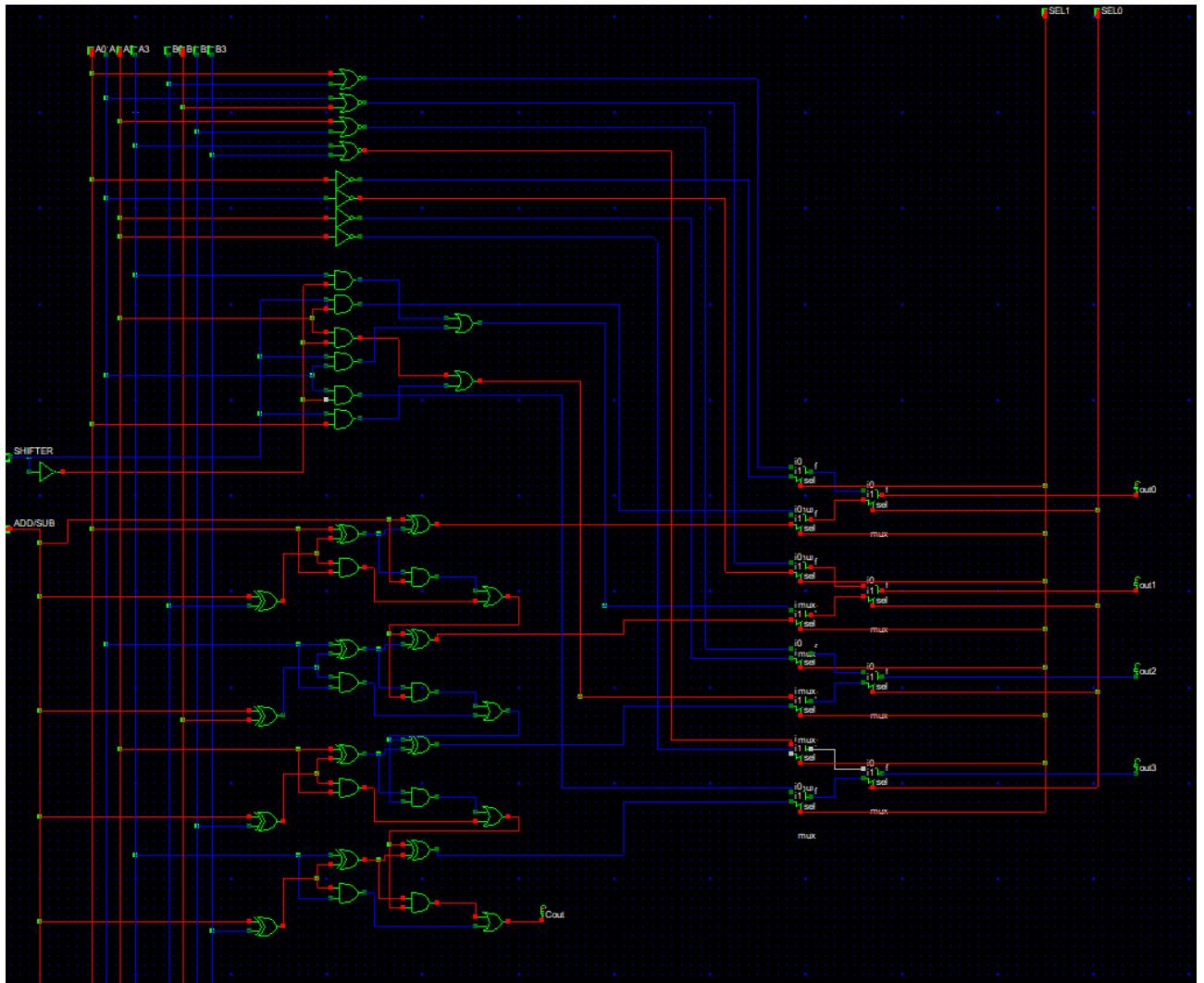


C. Simularea în DSCH

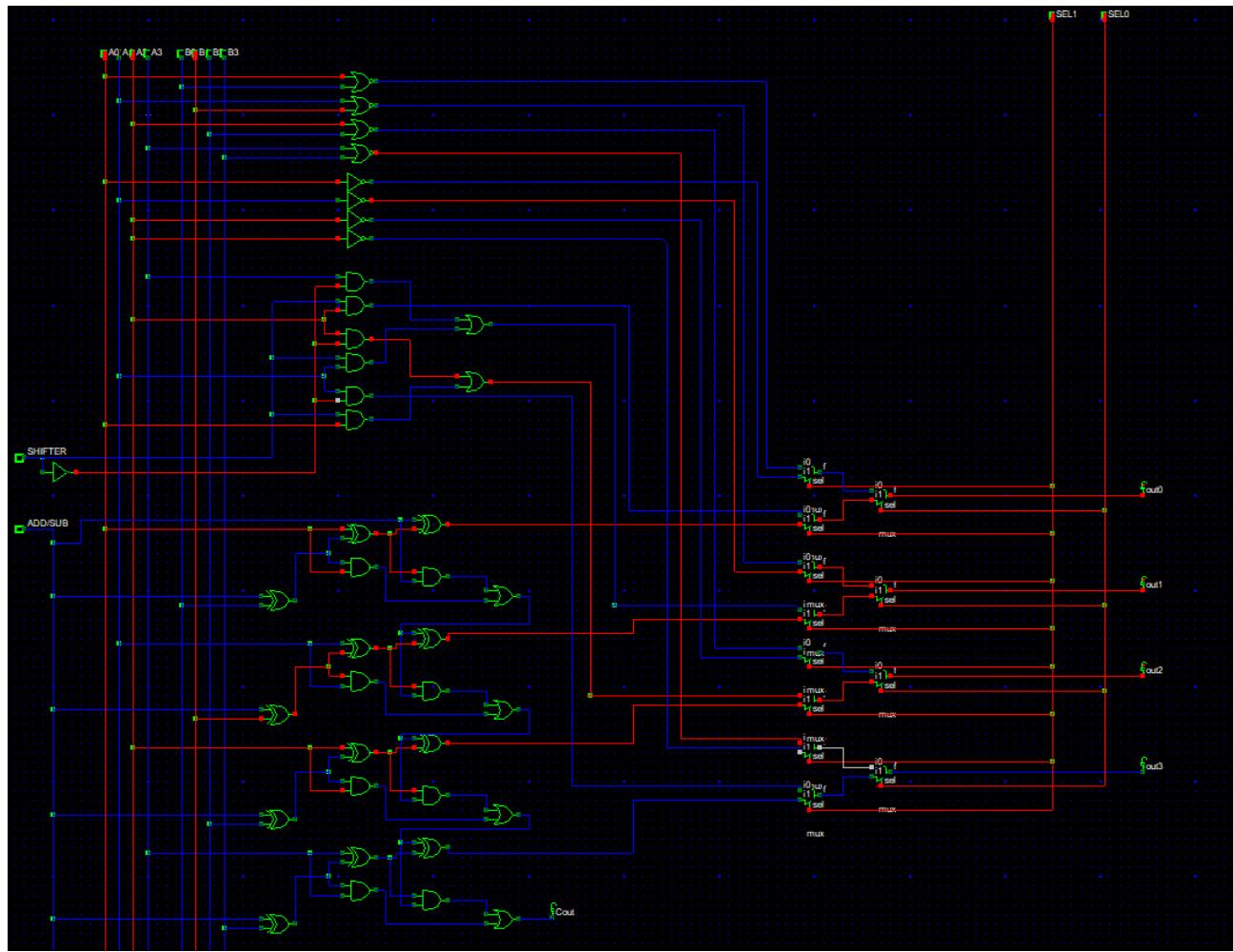
NOT



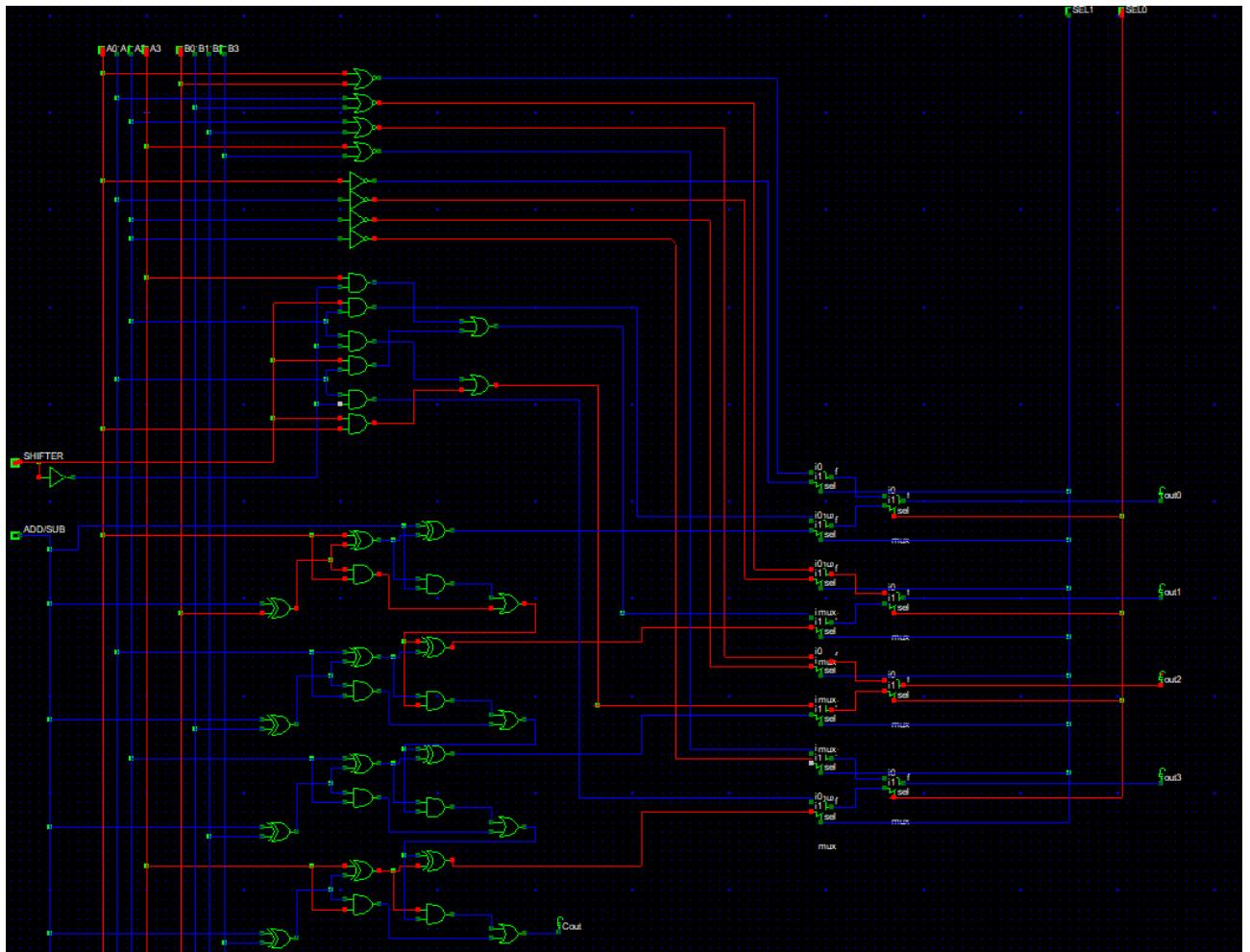
SUB

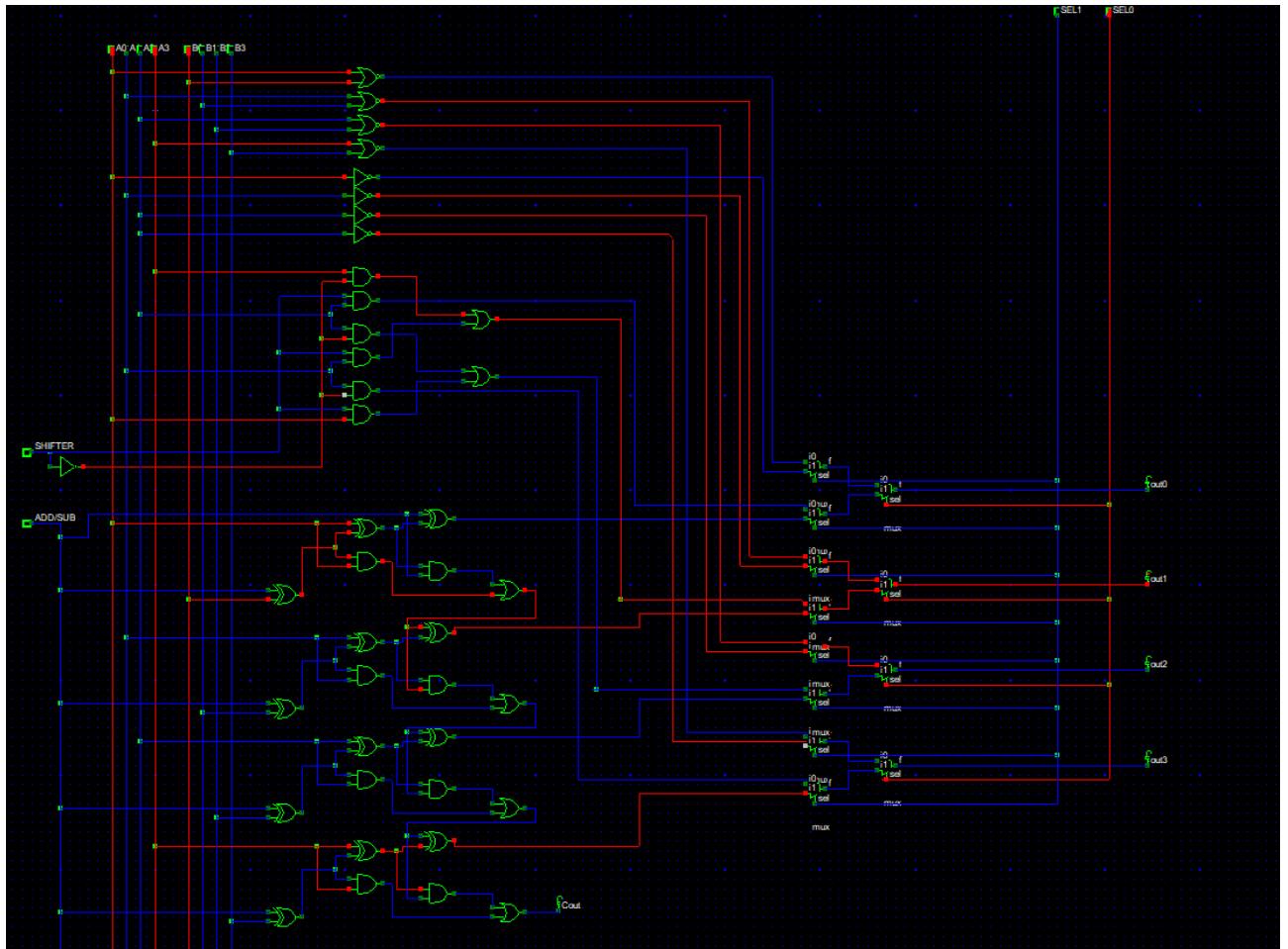


ADD



SHIFTER





D. Codul Verilog obținut

Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path |

```
// DSCH 3.5
// 5/12/2023 2:51:47 AM
// D:\VLSI\kituri_VLSI\dsch35_full\examples\ALU.sch

module ALU( SEL0,A1,A2,A3,B0,B1,B2,B3,
  SHIFTER,ADDbSUB,A0,SELL1,out3,Cout,out0,out1,
  out2);
  input SEL0,A1,A2,A3,B0,B1,B2,B3;
  input SHIFTER,ADDbSUB,A0,SELL1;
  output out3,Cout,out0,out1,out2;
  wire w10,w11,w12,w14,w15,w16,w17,w18;
  wire w19,w20,w21,w23,w24,w25,w26,w27;
  wire w28,w29,w31,w32,w34,w35,w36,w37;
  wire w38,w39,w40,w41,w42,w43,w44,w45;
  wire w46,w47,w48,w49,w50,w51,w52,w53;
  wire w54,w56,w57,w58,w60,w61,w63,w64;
  wire w66,w67,;
  not #(1) inv_1(w10,A2);
  not #(1) inv_2(w11,A2);
  not #(1) inv_3(w12,A1);
  not #(1) inv_4(w14,A0);
  nor #(3) nor2_5(w15,A2,B2);
  nor #(3) nor2_6(w16,A1,B1);
  nor #(3) nor2_7(w17,A0,B0);
  nor #(3) nor2_8(w18,A3,B3);
  or #(3) or2_9(w21,w19,w20);
  and #(3) and2_10(w20,A0,SHIFTER);
  and #(3) and2_11(w24,w23,A1);
  and #(3) and2_12(w25,A1,SHIFTER);
  and #(3) and2_13(w19,w26,A2);
  and #(3) and2_14(w27,A2,SHIFTER);
  and #(3) and2_15(w28,w26,A3);
  or #(3) or2_16(w29,w28,w25);
  not #(2) inv_17(w26,SHIFTER);
  or #(3) or2_18(Cout,w31,w32);
```

§ Verilog, Hierarchy and Netlist

Verilog | Hierarchy | Netlist | Critical path

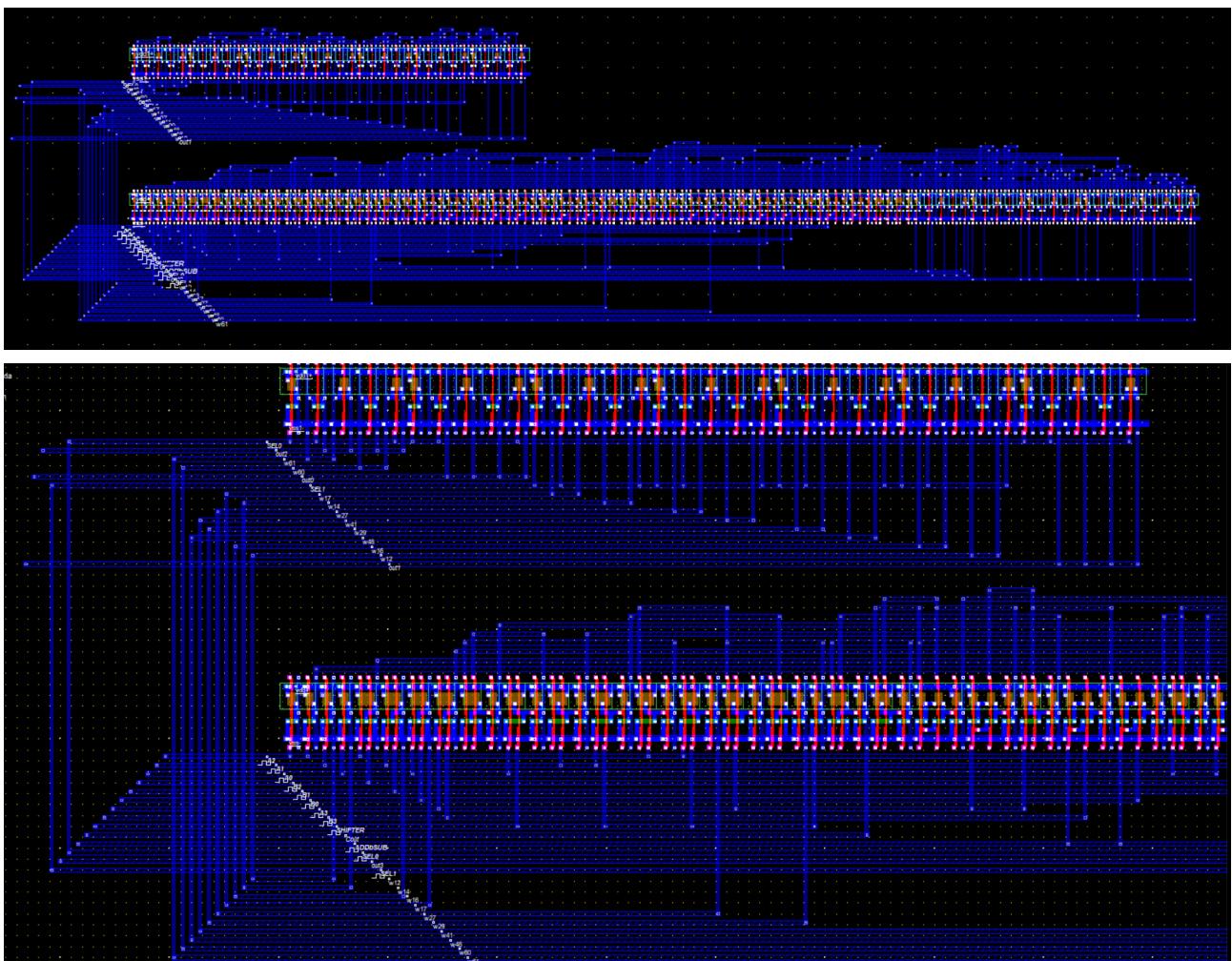
```
and #(3) and2_19(w31,w34,w35);
xor #(3) xor2_20(w36,w34,w35);
xor #(4) xor2_21(w35,A3,w37);
and #(3) and2_22(w32,A3,w37);
xor #(4) xor2_23(w37,ADDbSUB,B3);
xor #(4) xor2_24(w38,ADDbSUB,B0);
and #(3) and2_25(w39,A0,w38);
xor #(4) xor2_26(w40,A0,w38);
xor #(3) xor2_27(w41,ADDbSUB,w40);
and #(3) and2_28(w42,ADDbSUB,w40);
or #(4) or2_29(w43,w42,w39);
or #(4) or2_30(w46,w44,w45);
and #(3) and2_31(w44,w43,w47);
xor #(3) xor2_32(w48,w43,w47);
xor #(4) xor2_33(w47,A1,w49);
and #(3) and2_34(w45,A1,w49);
xor #(4) xor2_35(w49,ADDbSUB,B1);
xor #(4) xor2_36(w50,ADDbSUB,B2);
and #(3) and2_37(w51,A2,w50);
xor #(4) xor2_38(w52,A2,w50);
xor #(3) xor2_39(w53,w46,w52);
and #(3) and2_40(w54,w46,w52);
or #(4) or2_41(w34,w54,w51);
mux #(1) mux_42(out3,w56,w57,SEL0);
mux #(1) mux_43(w56,w18,w58,SEL1);
mux #(1) mux_44(w57,w24,w36,SEL1);
mux #(1) mux_45(w60,w21,w53,SEL1);
mux #(1) mux_46(w61,w15,w11,SEL1);
mux #(1) mux_47(out2,w61,w60,SEL0);
mux #(1) mux_48(out0,w63,w64,SEL0);
mux #(1) mux_49(w63,w17,w14,SEL1);
mux #(1) mux_50(w64,w27,w41,SEL1);
mux #(1) mux_51(w66,w29,w48,SEL1);
mux #(1) mux_52(w67,w16,w12,SEL1);
mux #(1) mux_53(out1,w67,w66,SEL0);
```

```
endmodule
```

```
// Simulation parameters in Verilog Format
always
#200 SEL0=~SEL0;
#400 A1=~A1;
#800 A2=~A2;
#1600 A3=~A3;
#3200 B0=~B0;
#6400 B1=~B1;
#12800 B2=~B2;
#25600 B3=~B3;
#51200 SHIFTER=~SHIFTER;
#102400 ADD/SUB=~ADD/SUB;
#102400 A0=~A0;
#102400 SEL1=~SEL1;

// Simulation parameters
```

E. Implementarea în Microwind



F. Simularea obținută în Microwind

Analog simulation of example

