

Rezumat

Acest laborator își propune introducerea în tehnologia Xilinx. Vor fi prezentați o serie de pași necesari în toate lucrările de laborator care vor urma.

Va fi trecut în revistă: etapele realizării unui design digital (descriere, simulare și prototipaj) folosind ca și produs software Xilinx ISE®. Circuitul descris este o poartă majority voting cu 3 intrări. Se va folosi Xilinx ISE® pentru sinteză, simulare post-sinteză și analiză de timp. Lucrările de laborator viitoare tratează circuite mai complexe. Scopul principal al acestui laborator este acela de a evidenția etapele de design și maniera concretă de realizare a acestora.

Obiectivele lucrării

Obiectivul acestui laborator este acela de cunoștere a software-ului pentru tehnologia Xilinx: Xilinx ISE® și folosirea acestuia în vederea realizării unui design simplu: majority voter cu 3 intrări.

Obiective tehnice

1. Familiarizare cu placa Nexys2
2. Realizarea unui design simplu: poartă majority folosind ISE® WebPACK™ și simularea lui.
3. Sinteza și implementare design.
4. Configurare FPGA Nexys2.

Timp necesar

2-3 ore

Pregătirea pentru laborator

- Citiți documentul înainte de a începe realizarea practică.
- Salvați output-urile pentru fiecare cerință sau anunțați instructorul în vederea prezentării rezultatelor.

Echipamente și Materiale

Acces la software-ul Xilinx

Necesar	Cantitate
Software ISE® WebPACK™ 14.4 de pe pagina de WEB Xilinx, www.xilinx.com .	1
Plugin Digilent (www.digilent.com)	1
Placă Digilent Nexys 2	1

Alte Referințe

Manualul de Xilinx ISE de la pagina de WEB Xilinx: www.xilinx.com.



Activitate 1: Placa Digilent Nexys2

Efectuați unul dintre pași 1.1, respectiv 1.2 funcție de sistemul de operare existent.

Pas 1.1: Instalare plug-in Digilent pentru sisteme Windows cu Adept

Descărcați și instalați tool-ul Adept de pe site-ul Digilent și instalați-l.

Adept 2.15.3 System, 32/64-bit Window -

<http://www.digilentinc.com/Products/Detail.cfm?Prod=ADEPT2>

Pas 1.2: Instalare plug-in Digilent pentru sisteme Linux

Descărcați Adept Runtime și Adept Utilities de pe site-ul Digilent.

Adept 2.15.3 Runtime, X86 Linux

1. Dezarhivați:

```
gzip -d digilent.adept.runtime_2.15.3-i686.tar.gz
```

2. Extrageți fișierele:

```
tar -xvf digilent.adept.runtime_2.15.3-i686.tar
```

3. Din directorul unde sunt dezarhivate fișierele:

```
sudo ./install.sh
```

4. Din directorul unde sunt dezarhivate fișierele:

```
sudo ./install.sh
```

 și tastați două ENTER-uri consecutive

Adept 2.15.3 Runtime, X86 Linux

1. Dezarhivați:

```
gzip -d digilent.adept.utilities_2.1.1-i686.tar.gz
```

2. Extrageți fișierele:

```
tar -xvf digilent.adept.utilities_2.1.1-i686.tar
```

3. Din directorul unde sunt dezarhivate fișierele:

```
sudo ./install.sh
```

4. Din directorul unde sunt dezarhivate fișierele:

```
sudo ./install.sh
```

 și tastați două ENTER-uri consecutive

Verificați funcționarea corespunzătoare tastând:

```
djtgcfg
```

```
djtgcfg enum
```

Placa Digilent trebuie să fie conectată și trebuie să poată fi detectată.

Logica Digitala

Lab1: Intro Xilinx ISE®



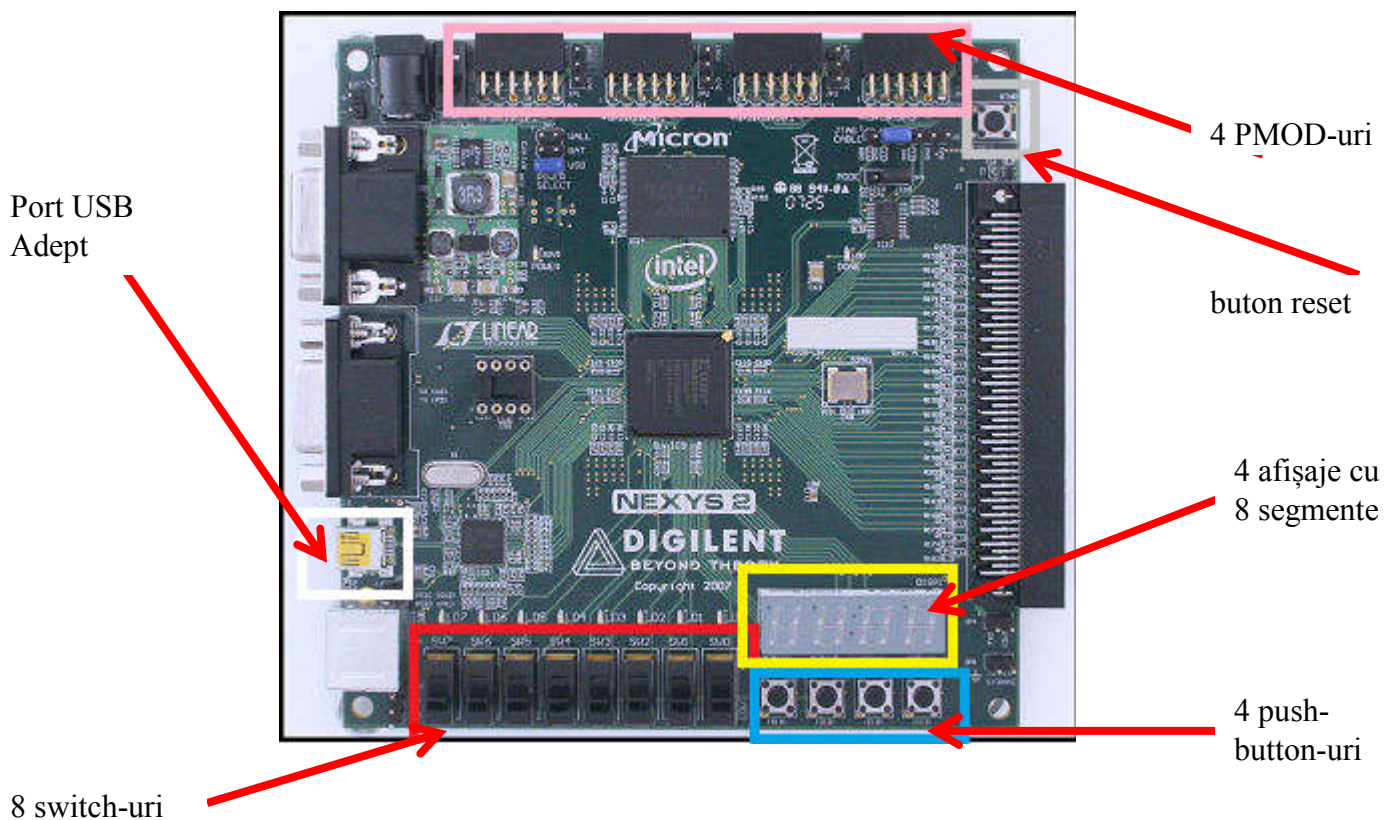
Pas 2: Introducere placă Digilent Nexys2

Documentația aferentă plăcii Nexys2 o găsiți pe site-ul Digilent:

<http://www.digilentinc.com/Products/Detail.cfm?NavTop=2&NavSub=451&Prod=NEXYS2&CFID=4081352&CFTOKEN=fe3ecd12e62db420-F51AFE65-5056-0201-0230536D512CABC6>

Dintre caracteristicile plăcii menționăm:

- Xilinx Spartan-3E FPGA, 500K porți
- port USB2 care asigură alimentarea porții, configurarea dispozitivului, și transfer date de mare viteză
- Poate fi folosită în conjuncție cu ISE/Webpack și EDK
- 16MB Micron PSDRAM , Flash
- tact 50MHz
- 75 FPGA I/O ruți la conectori de expansiune (conector Hirose FX2 connector de mare viteză cu 43 semnale și 4 2x6 Pmod conectori)
- I/O ale plăcii care include: 8 LED-uri (marcate cu roșu), 4-afișaje cu 8 segmente, 4 pushbutton-uri, 8 switch-uri

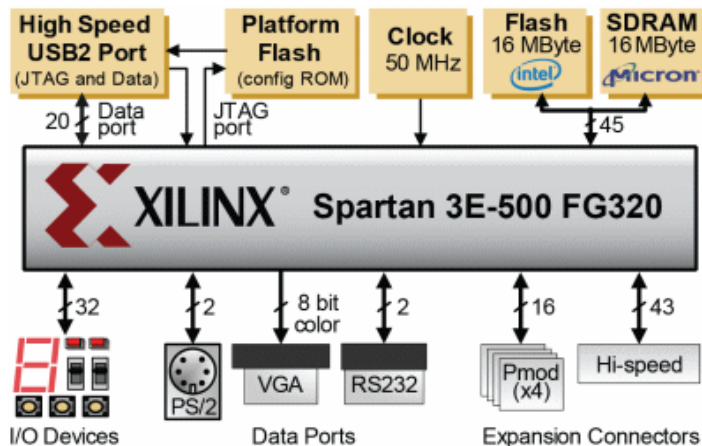


Logica Digitala

Lab1: Intro Xilinx ISE®

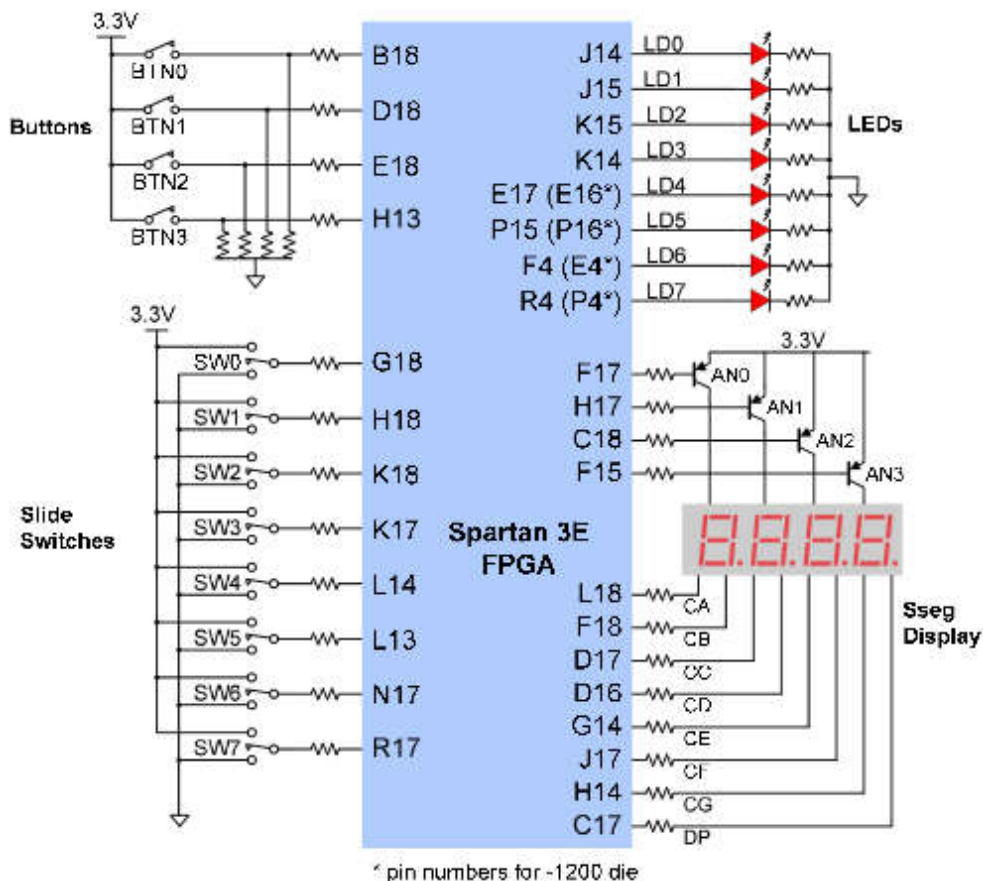


Schema plăcii este prezentată în figura următoare (imaginile plăcii Nexys2 sunt preluate de pe site-ul Digilent).



Pentru mai multe informații se recomandă consultarea documentului Nexys2 Reference manual: [aici](#).

În continuare vom trece în revistă facilitățile de I/O oferite de placa Nexys2:



Logica Digitala

Lab1: Intro Xilinx ISE®



Intrări:

- Butoane care se apasă (pushbutton): aceste intrări sunt în mod normal la 0 logic și se setează la 1 logic numai în momentul în care sunt apăstate.
- Întrerupătoare (switch-uri): acestea generează semnale de intrare constante 0 sau 1 logic funcție de poziția întrerupătorului.

Ieșiri:

- LED-urile: un led va fi aprins dacă semnalul de ieșire conectat la acesta este 1 logic, altminteri el este stins.
- 4 afișaje cu 7 segmente: acestea sunt multiplexate. Revenim cu mai multe informații într-o lucrare viitoare când ne propunem realizarea unui decodificator pentru un astfel de afișaj.



Activitate 2: Realizarea design-ului pentru o poartă majority voting cu 3 intrări

Etapele realizării unui Design folosind tehnologia Xilinx:

Pas 1: Descrierea design-ului

- Două modalități:
 1. Limbaje de descriere hardware HDL (Verilog sau VHDL)
 2. Folosind scheme (editorul de scheme).

Această lucrare ilustrează ambele modalități: descriere Verilog HDL și editorul de scheme.

Pas 2: Sinteza design-ului

- Translatarea codului Verilog HDL și/sau fișierelor cu scheme într-un format standard – fișier EDIF.

Pas 3: Implementarea design-ului

- Translatare, Mapare pe componentele FPGA-ului, Alocarea componentelor specifice dispozitivului FPGA, și rutare în vederea stabilirii interconexiunilor dintre componente. Ieșirea acestui process este un fișier (.BIT) folosit pentru programarea FPGA-ului.

Pas 4: Configurare echipament Xilinx

- Descărcarea fișierului pe FPGA

Acest laborator prezintă modul în care poate fi realizat design-ul unui circuit digital simplu: majority voter cu 3 intrări folosind Xilinx ISE. În acest laborator sunt realizați pașii 1 la 4 din procesul de design prezentat anterior. Reamintim faptul că pașii 1 și 2 au constituit obiectivul laboratorului trecut. Pentru mai multe detalii accesați prima parte a acestui tutorial.

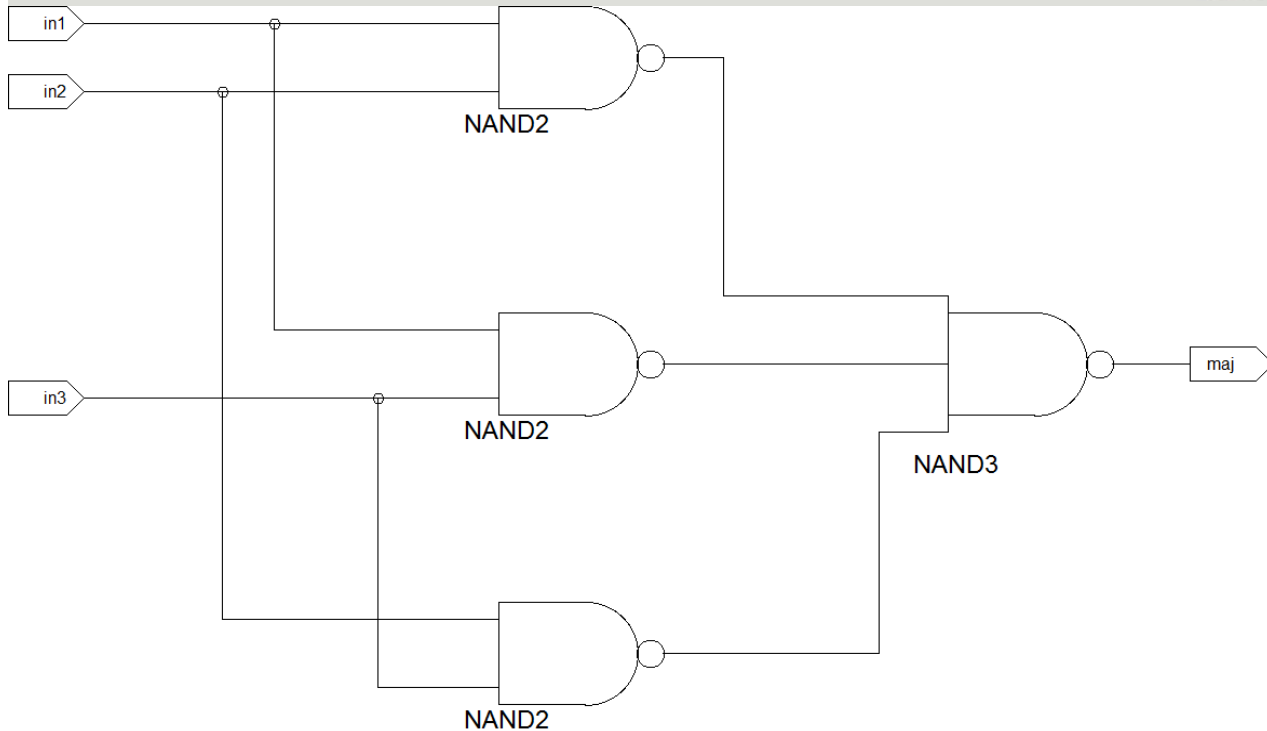
Pasul 1: Crearea unui proiect ISE nou și crearea design-ului unei componente majority voting

Succint vor fi punctate etapele realizării unui proiect nou:

- Pentru pornire ISE: deschideți un terminal și tastați *ise*
- Creați un proiect nou în directorul workspace: MajVoter
 - **File → New Project.** se deschide **new project wizard**.
 - Selectați corespunzător locația unde doriți să creați proiectul;
 - Top level source type: HDL;
 - Opțiunile proiectului:
 - **Spartan-3E Starter Board** la **Evaluation Development Board**;
 - **Verilog** la **Preferred Language**;
- În continuare realizați fie printr-un schematic, fie cod Verilog următoarea descriere de componentă:

Logica Digitala

Lab1: Intro Xilinx ISE®



- La **Hierarchy** în tab-ul de **Design** selectați **Project**→**New source** deschide fereastra **New Source Wizard**
 - Pentru implementarea folosind schematic alegeți la **Select Source Type** – **Schematic**
 - Pentru implementarea folosind descrierea Verilog HDL alegeți la **Select Source Type** – **Verilog Module**

Pentru opțiunea cu Verilog module codul aferent este următorul. Subliniat găsiți liniile de cod care trebuie adăugate pentru a modela comportamentu unității majority voting.

```
module majority_voter(  
    input in1,  
    input in2,  
    input in3,  
    output maj  
);
```

```
    wire nand1, nand2, nand3; //internal wires
```

```
    assign nand1 = ~(in1 & in2);  
    assign nand2 = ~(in1 & in3);  
    assign nand3 = ~(in2 & in3);
```


Logica Digitala

Lab1: Intro Xilinx ISE®



```
assign maj = ~(nand1 & nand2 & nand3);
```

```
endmodule
```

- Adăugați la proiect un fișier de tip testbench :
 - **Project** → **New source** deschide fereastra **New Source Wizard**, alegeți la **Select Source Type** – **Verilog Test Fixture**

Fișierul testbench este următorul. Sunt scose în evidență liniile de cod adăugate pentru generarea stimulilor de intrare.

```
module majority_voter_tb;

    // Inputs
    reg in1;
    reg in2;
    reg in3;

    // Outputs
    wire maj;

    // Instantiate the Unit Under Test (UUT)
    majority_voter uut (
        .in1(in1),
        .in2(in2),
        .in3(in3),
        .maj(maj)
    );

    initial begin
        // Initialize Inputs
        in1 = 0;
        in2 = 0;
        in3 = 0;

        // Wait 100 ns for global reset to finish
        #100;

    end
    always //toggle inputs
    begin
        #25 in1 = ~in1;
        #50 in2 = ~in2;
```


Logica Digitala

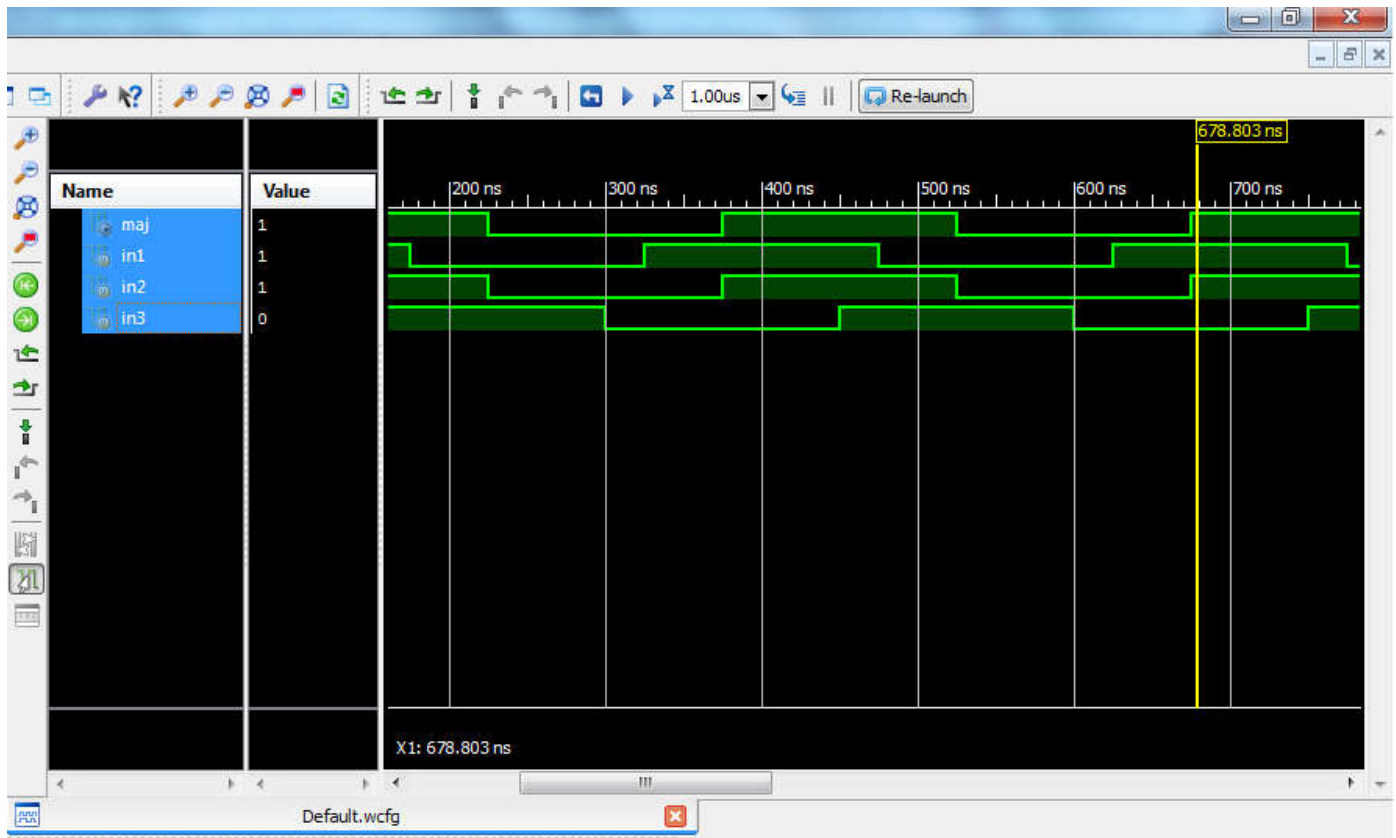
Lab1: Intro Xilinx ISE®



```
#75 in3 = ~in3;
end
```

endmodule

- Simulați circuitul folosind ISim-ul. Ieșirea obținută în urma simulării se prezintă astfel:



- Extrageți tabelul de adevăr pentru componenta modelată. Pentru toate combinațiile de valori de intrare, extrageți din simulare valoarea obținută la ieșire.

in1	in2	in3	maj
0	0	0	0
0	0	1	0
...	?
1	1	1	?

Logica Digitala

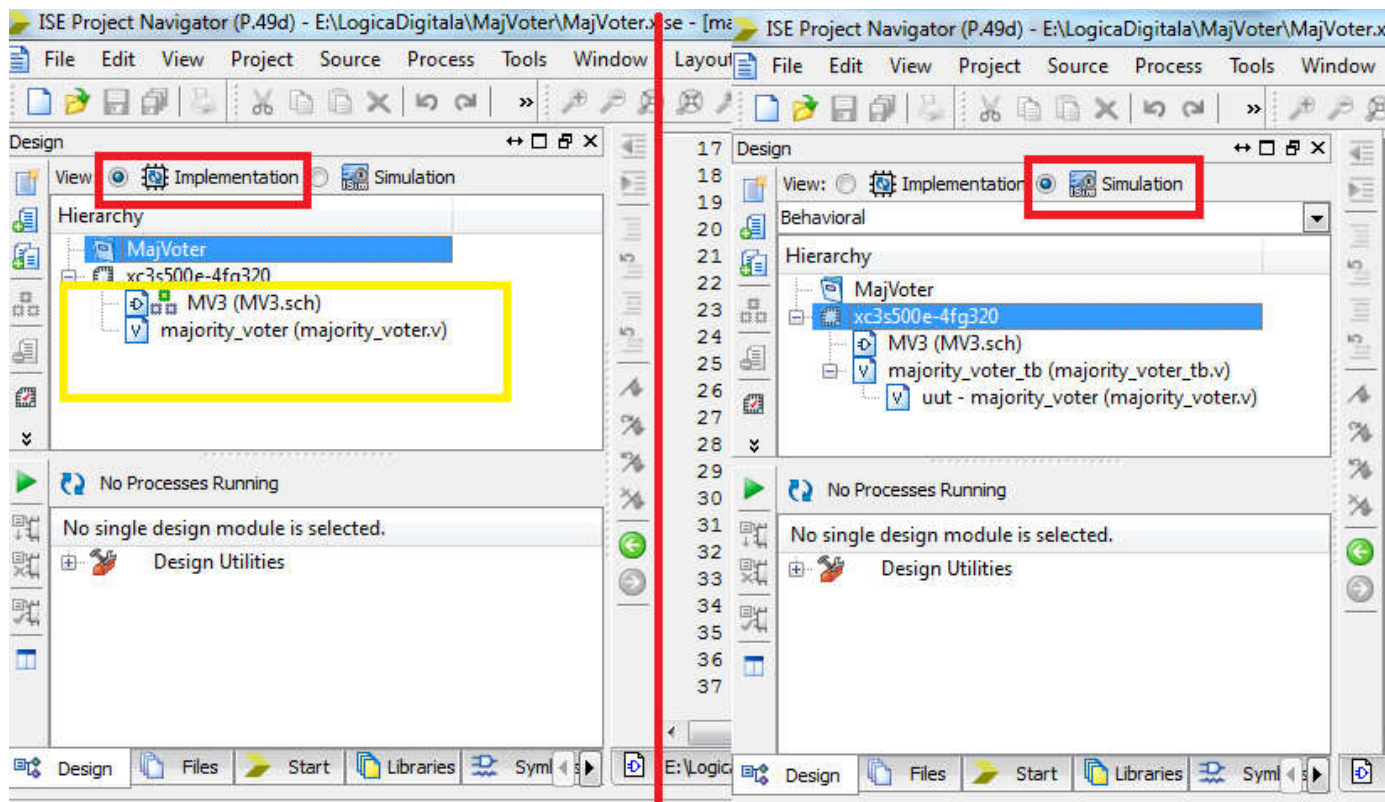
Lab1: Intro Xilinx ISE®



Pasul 2: Sinteză design

La **Hierarchy** în tab-ul de **View** selectați **Implementation**.

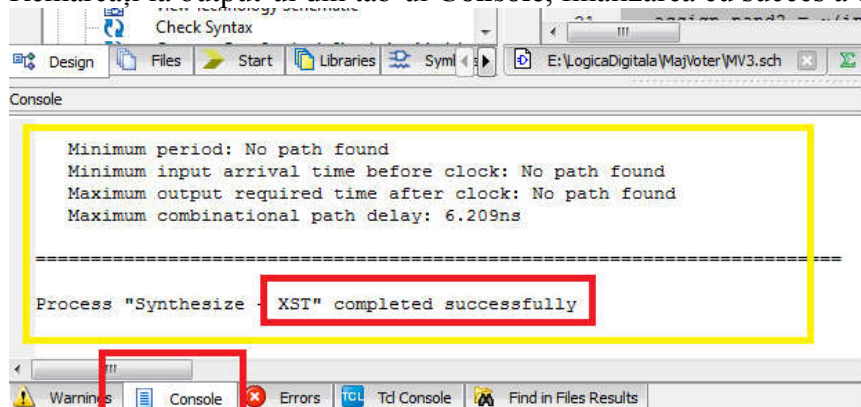
Remarcați faptul că fișierul testbench a dispărut.



În continuare selectați modulul care doriți să-l setați ca și top-level (cel al cărui design va fi programat pe FPGA).

În tab-ul de Design dați click pe **Synthesize→Run**. Alternativa este să dați dublu click pe Synthesize.

Remarcați la output-ul din tab-ul **Console**, finalizarea cu succes a operației de sinteză.



Logica Digitala

Lab1: Intro Xilinx ISE®



Pasul 3: Implementare design-ului

Înainte de a trece la configurarea design-ului pe placă mai avem nevoie de realizarea unui fișier care să specifice atribuirile pentru pinii de I/O la porturile de I/O din design-ul care se dorește verificat pe FPGA. Pentru a scrie informațiile din UCF este necesară cunoașterea plăcii folosite. În acest laborator ea este Nexys2 cu FPGA-ul Spartan3-E 500 FG320. Toate aceste informații se găsesc specificate în manualul plăcii (Board Reference Manual).

Circuitul pe care dorim să-l verificăm folosește 3 comutatoare pentru intrări și un led pentru ieșiri.

SW0 – pin G18,

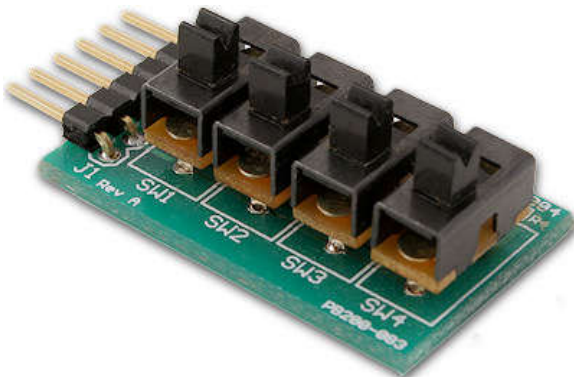
SW1 – pin H18,

SW2 – pin K18,

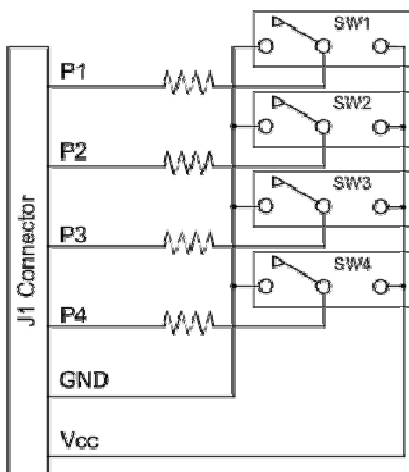
LED0 – pin J14.

Dacă vrem o componenta PmodSWT care este conectată la interfața PMOD1, atunci trebuie consultat manualul aferent acestuia și trebuie identificați pinii pentru conectorul PMOD1 al plăcii Digilent Nexys2.

Conform manualului componenta PmodSWT se prezintă astfel:



Cu schema:



Logica Digitala

Lab1: Intro Xilinx ISE®

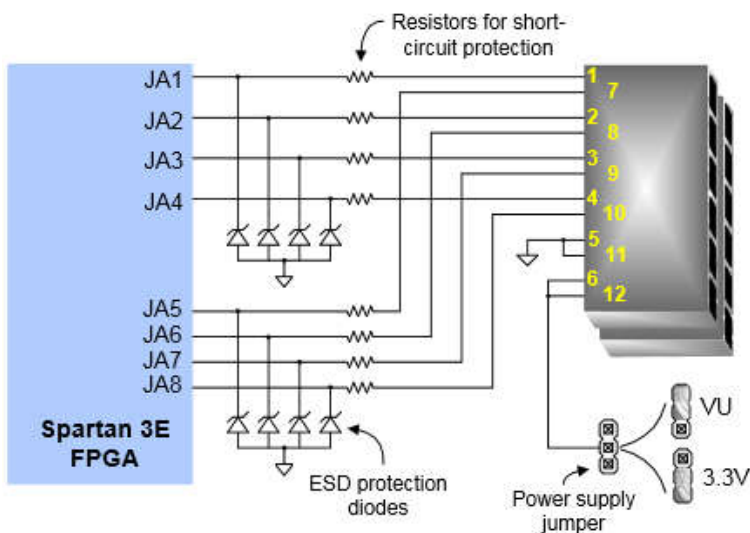


Aceste nume, împreună cu pinii aferenți mai pot fi identificați:

- De pe placă;
- Din Master UCF;

Pentru placa Nexys2, din manual studiați specificația pentru PMOD1 și extrageți informațiile referitoare la pini.

Placa oferă 4 conectori x 2 rânduri de câte 6 pini fiecare. Astfel până la 8 plăci de expansiune PMOD pot fi conectate. Fiecare grup de 8 pini are 8 aferenți semnalelor de date, 2 pini pentru masă (GND) și 2 pini pentru alimentare (Vdd). Conexiunile fizice sunt prezentate în imaginea următoare.



Iar pinii în tabelul de mai jos:

Table 3: Nexys2 Pmod Connector Pin Assignments

Pmod JA		Pmod JB		Pmod JC		Pmod JD	
JA1: L15	JA7: K13	JB1: M13	JB7: P17	JC1: G15	JC7: H15	JD1: J13	JD7: K14 ¹
JA2: K12	JA8: L16	JB2: R18	JB8: R16	JC2: J16	JC8: F14	JD2: M18	JD8: K15 ²
JA3: L17	JA9: M14	JB3: R15	JB9: T18	JC3: G13	JC9: G16	JD3: N18	JD9: J15 ³
JA4: M15	JA10: M16	JB4: T17	JB10: U18	JC4: H16	JC10: J12	JD4: P18	JD10: J14 ⁴

Notes: ¹ shared with LD3 ² shared with LD3 ³ shared with LD3 ⁴ shared with LD3

Așadar pentru situația când conectăm un PMOD cu 4 comutatoare pinii selectați sunt:

SW0 – pin L15	#SW1 PMODSWT conectat la grupul A
SW1 – pin K12	#SW2 PMODSWT conectat la grupul A
SW2 – pin L17	#SW3 PMODSWT conectat la grupul A
LED0 – pin J14	#Led0 Placă Nexys2

Observație:

Logica Digitala

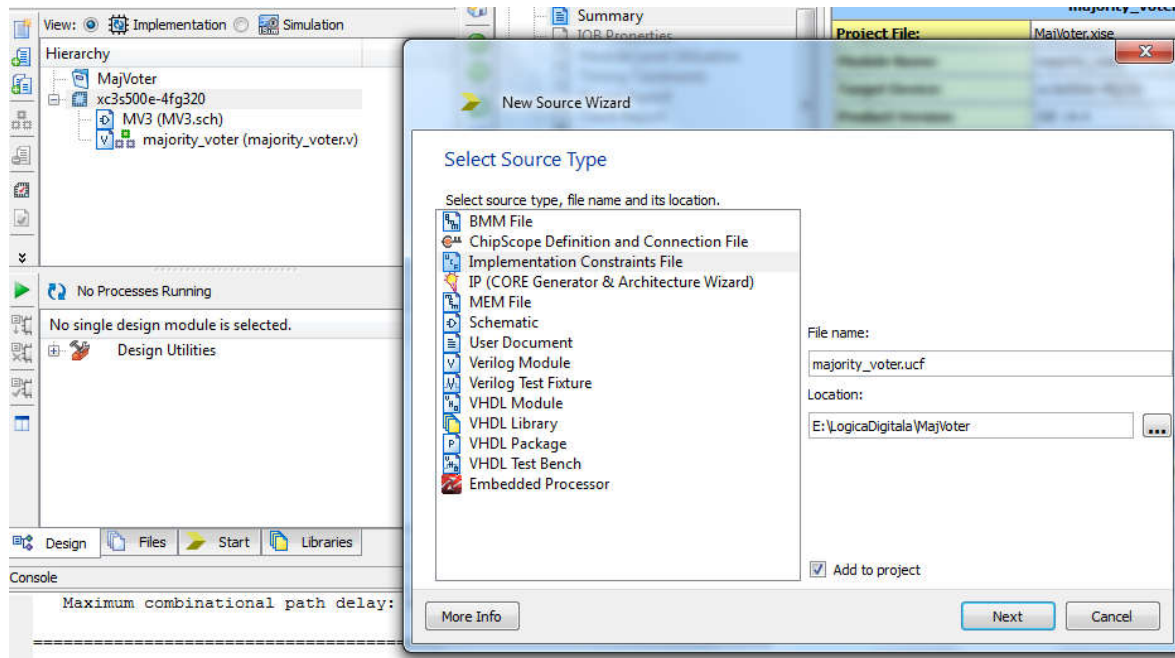
Lab1: Intro Xilinx ISE®



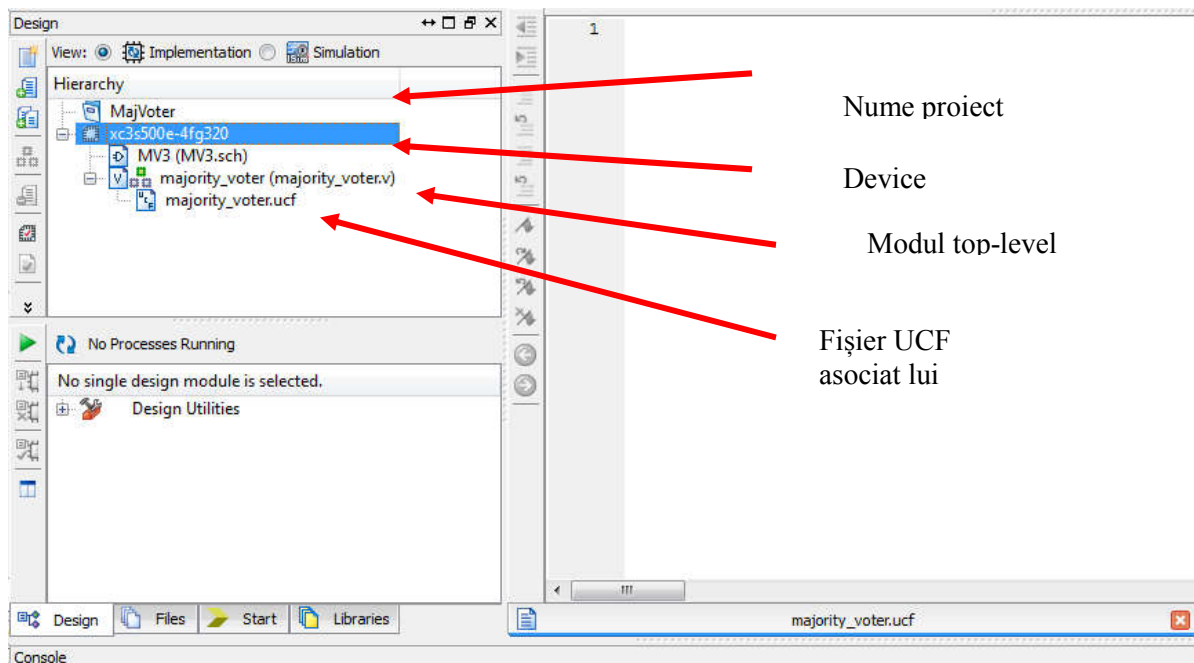
Din punct de vedere al sintaxei, în UCF un comentariu începe cu caracterul #.

Pentru adăugarea fișierului UCF urmăriți secvența de operații:

- **Right-click** pe Hierarchy Pane și selectați **New Source**, după care alegeți la tipul sursei **Implementation Constraints File**. Denumiți fișierul astfel creat. De regulă este același nume ca și cel al modulului descărcat pe placă.



Astfel un fișier UCF gol este adăugat.



Logica Digitala

Lab1: Intro Xilinx ISE®



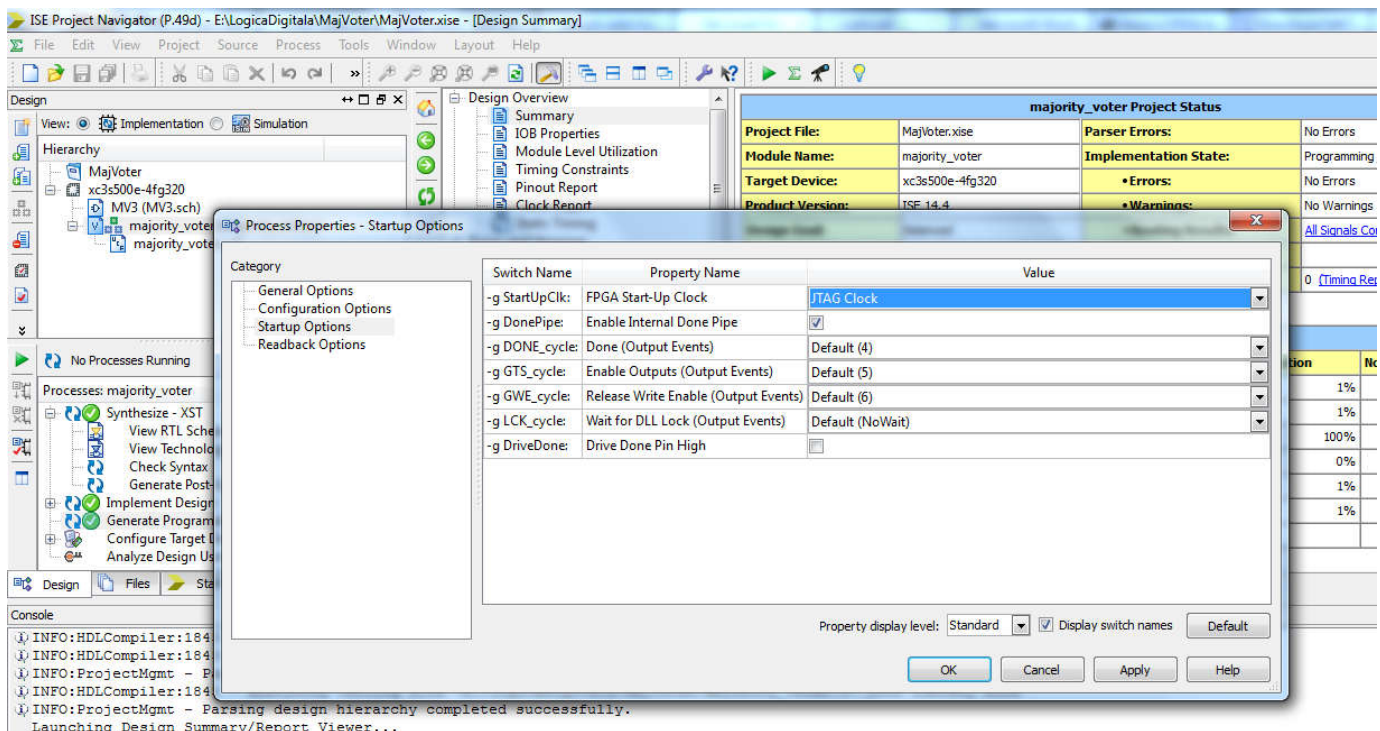
Acesta se completează cu informațiile extrase legate de pini.

```
1 #Inputs
2 NET "in1" LOC = "L15"; # SW1 PMODSW1 conectat la grupul A
3 NET "in2" LOC = "K12"; # SW2 PMODSW2
4 NET "in3" LOC = "L17"; # SW3 PMODSW3
5
6 # Outputs
7 NET "maj" LOC = "J14"; # LED 0 for majority voter output - Placa Nexys2
8
9
```

Numele porturilor modulului top-level la care este asociat fișierul UCF

Pasul următor constă din crearea fișierului de configurare *.bit.

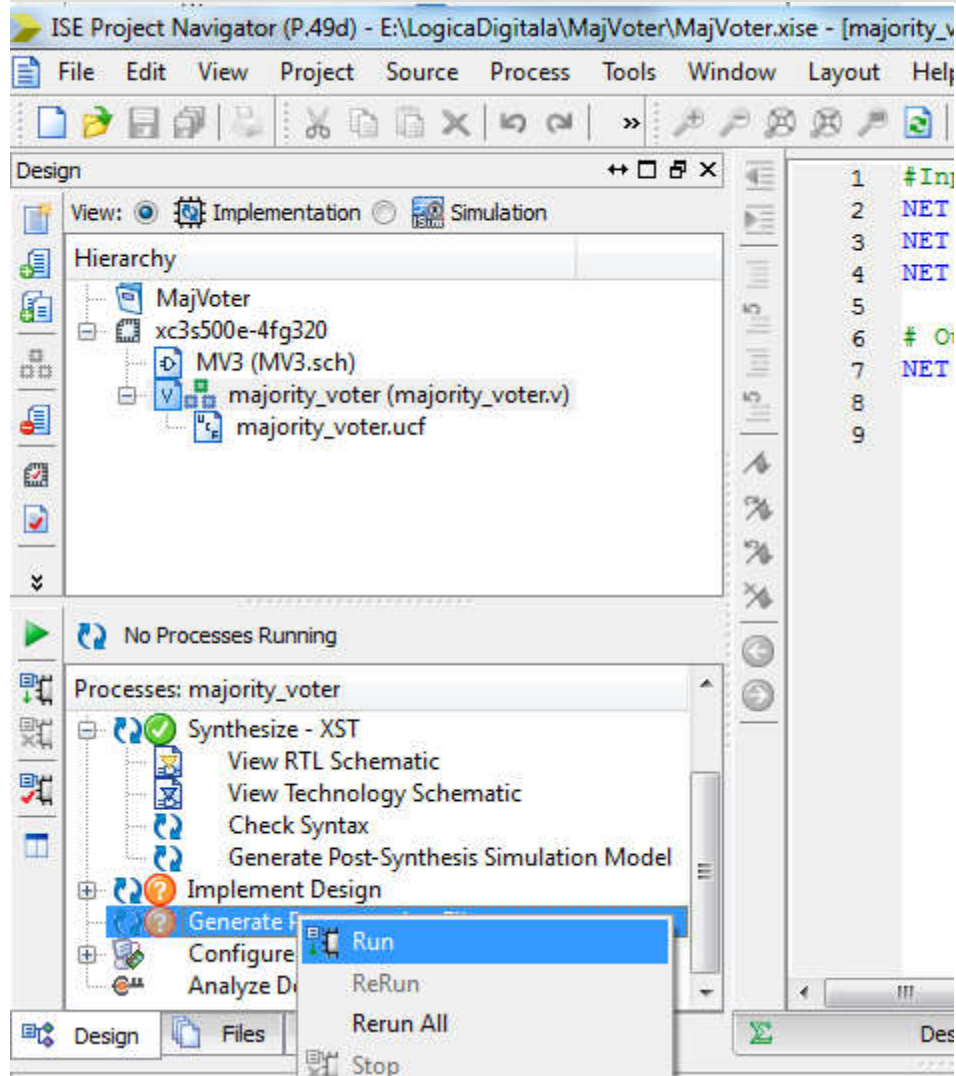
Înainte de a genera fișierul *.bit, dați click dreapta pe Generate Programming File și selectați Process Properties, apoi meniul Startup Options. FOARTE IMPORTANT: Modificați default FPGA start-up clock din CCLK în JTAG.



În continuare în tab-ul de **Design** dați click pe **Generate Programming File** și apoi **Run**.

Logica Digitala

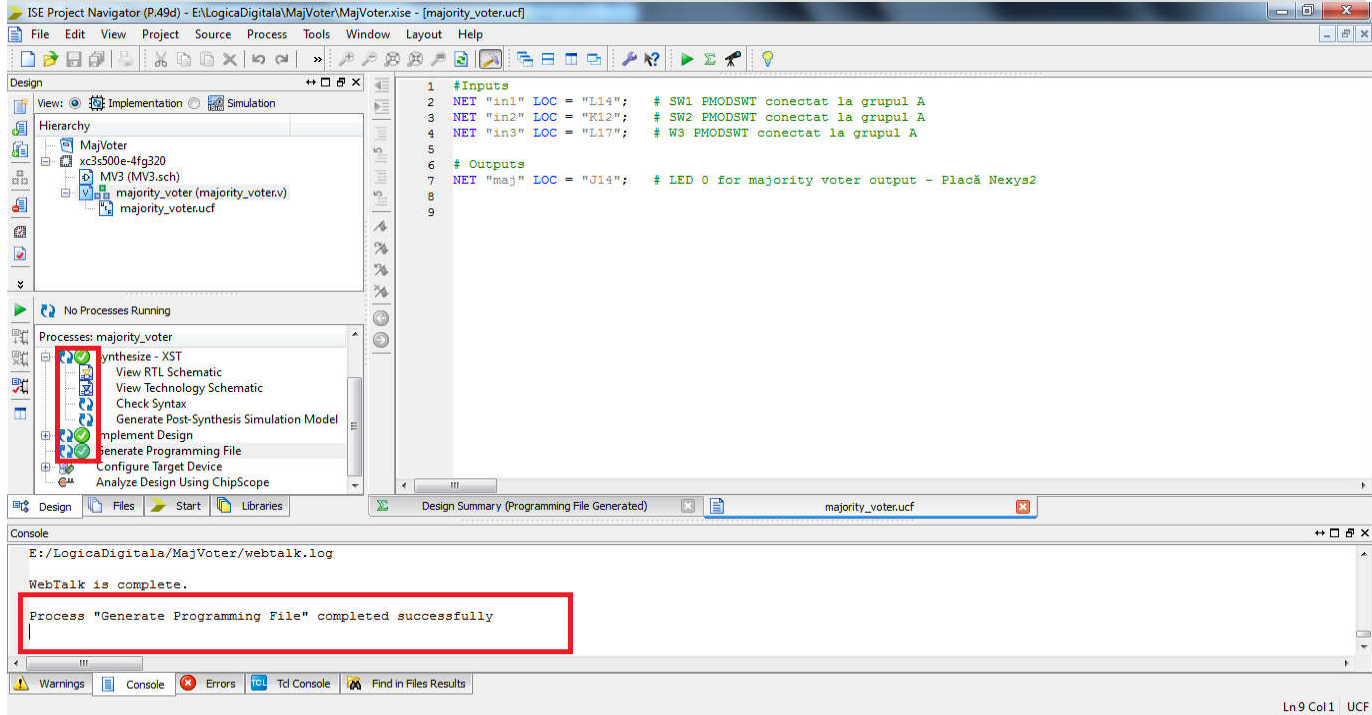
Lab1: Intro Xilinx ISE®



Fișierul *.bit a fost astfel generat.

Logica Digitala

Lab1: Intro Xilinx ISE®



Pasul 4: Configurare placă Digilent Nexys2

Ultimul pas constă în descărcarea design-ului pe placă.

Din Terminal tastați:

```
djtgcfg prog -d Nexys2 -i 0 -f majority_voter.bit
```

Verificați funcționarea corectă a design-ului!

Logica Digitala

Lab1: Intro Xilinx ISE®



Bibliografie:

1. Xilinx docs. www.xilinx.com
2. Craig Kief,, Alonzo Vera, Alexandria Haddad, and Quinlan Cao. Xilinx ISE tutorials.
3. John F. Wakerly – „Digital Design: Principles and Practices”, 3rd Edition, Prentice Hall, 2000, ISBN 0-13-769191-2
4. J. Bhasker A Verilog HDL Primer, Third Edition Star Galaxy Publishing, 2005