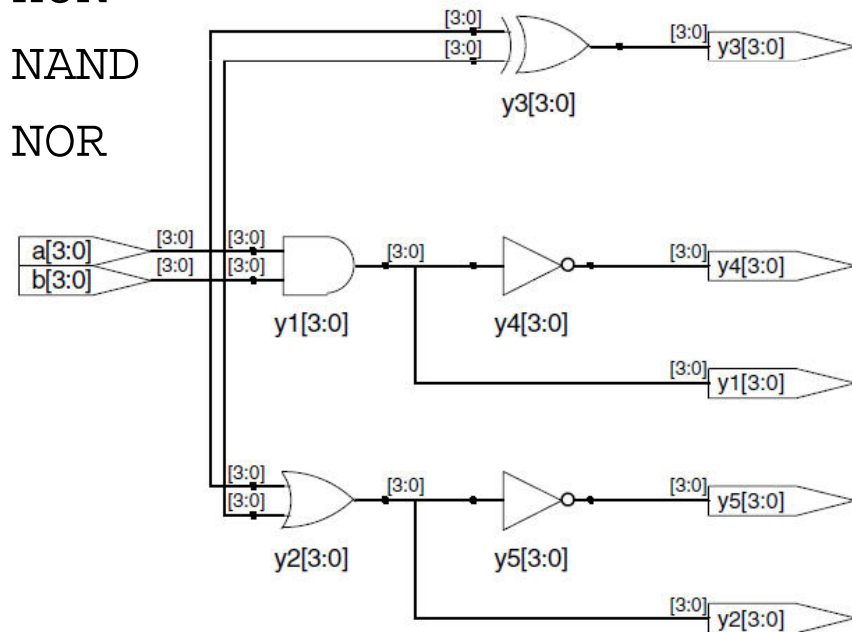
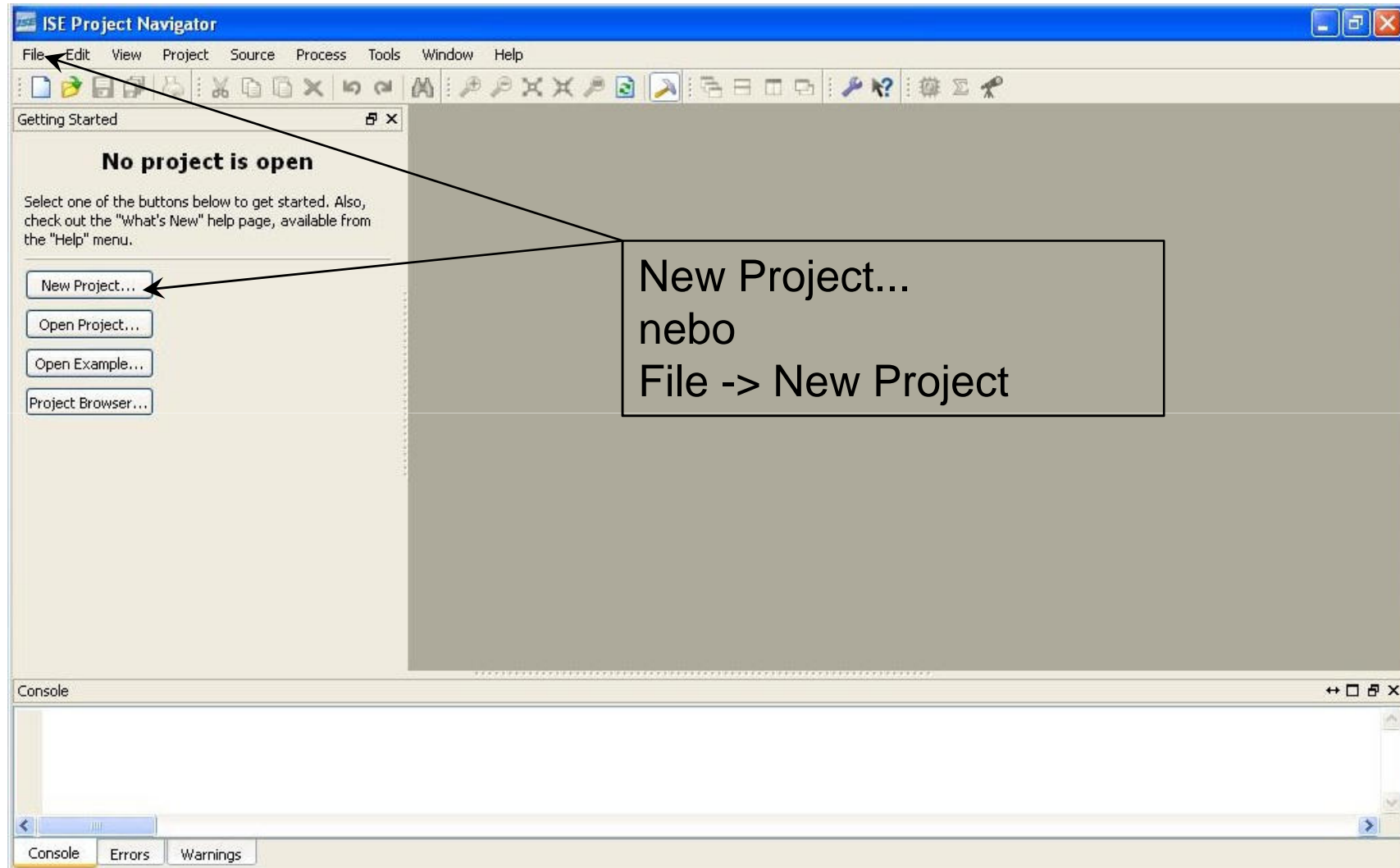


## Verilog – příklad – behaviorální popis

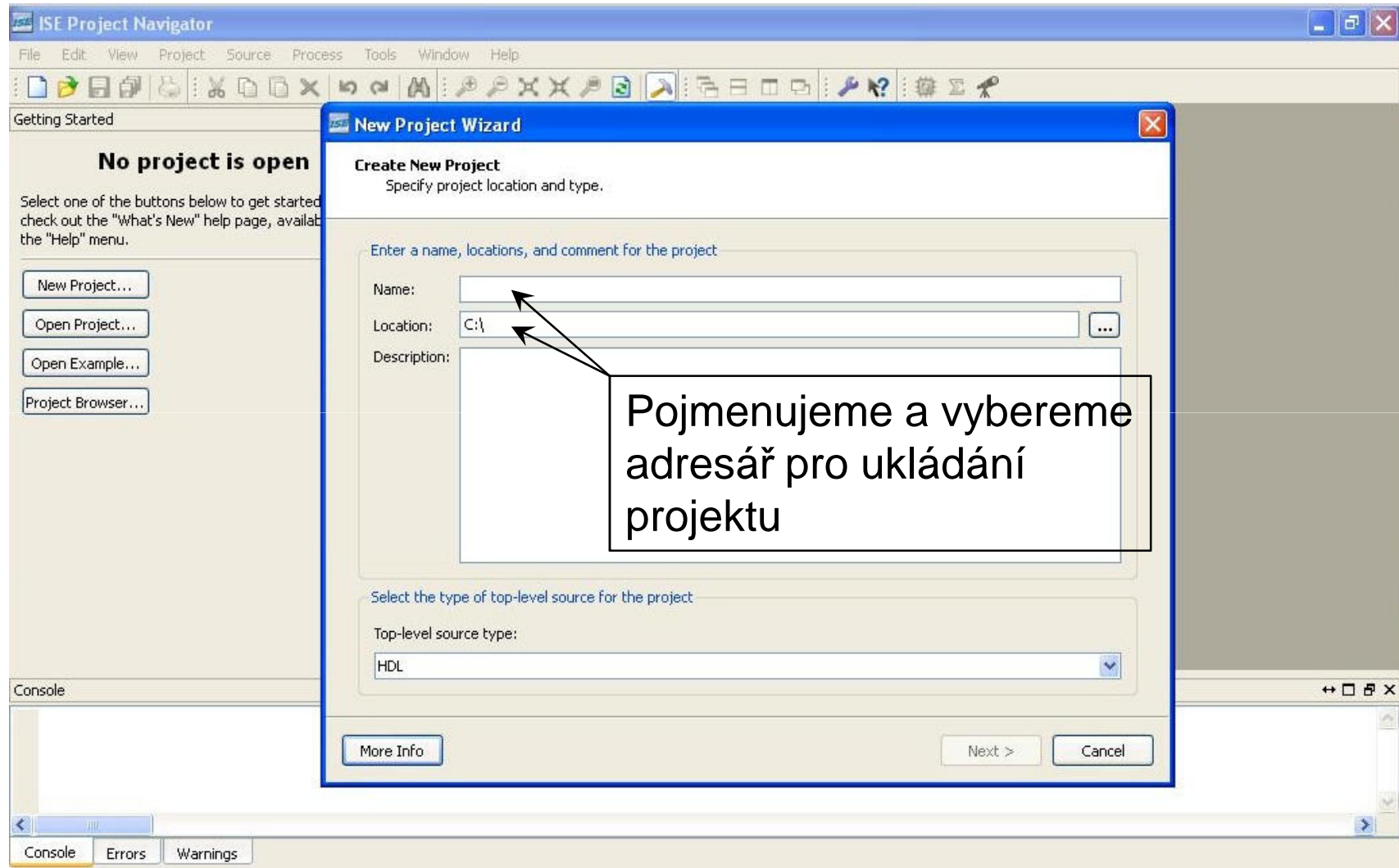
```
module gates (input [3:0] a, b,  
              output [3:0] y1, y2, y3, y4, y5);  
  
  assign y1 = a & b;           // AND  
  assign y2 = a | b;          // OR  
  assign y3 = a ^ b;          // XOR  
  assign y4 = ~(a & b);       // NAND  
  assign y5 = ~(a | b);       // NOR  
  
endmodule
```



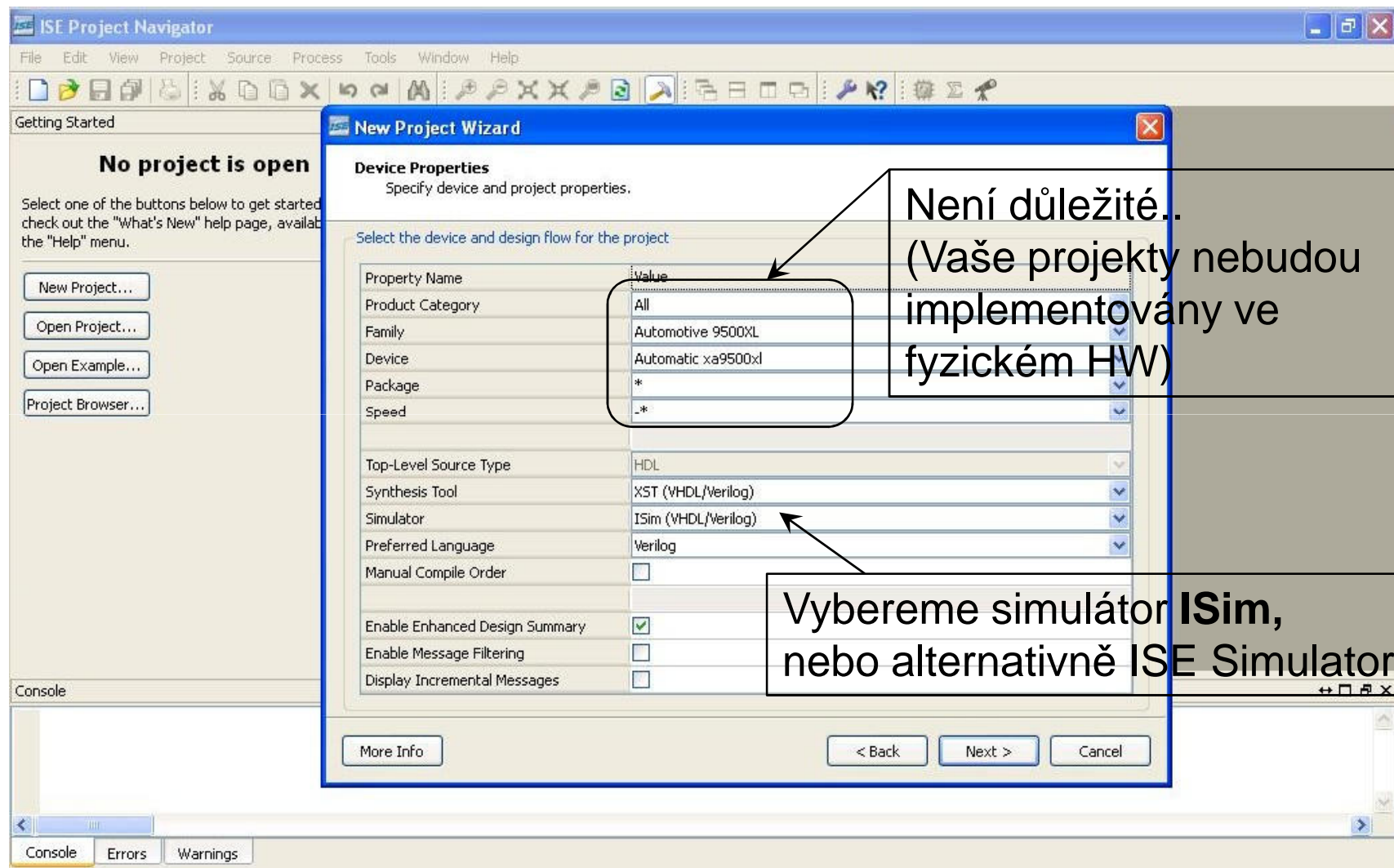
# Založení nového projektu v prostředí Xilinx IDE



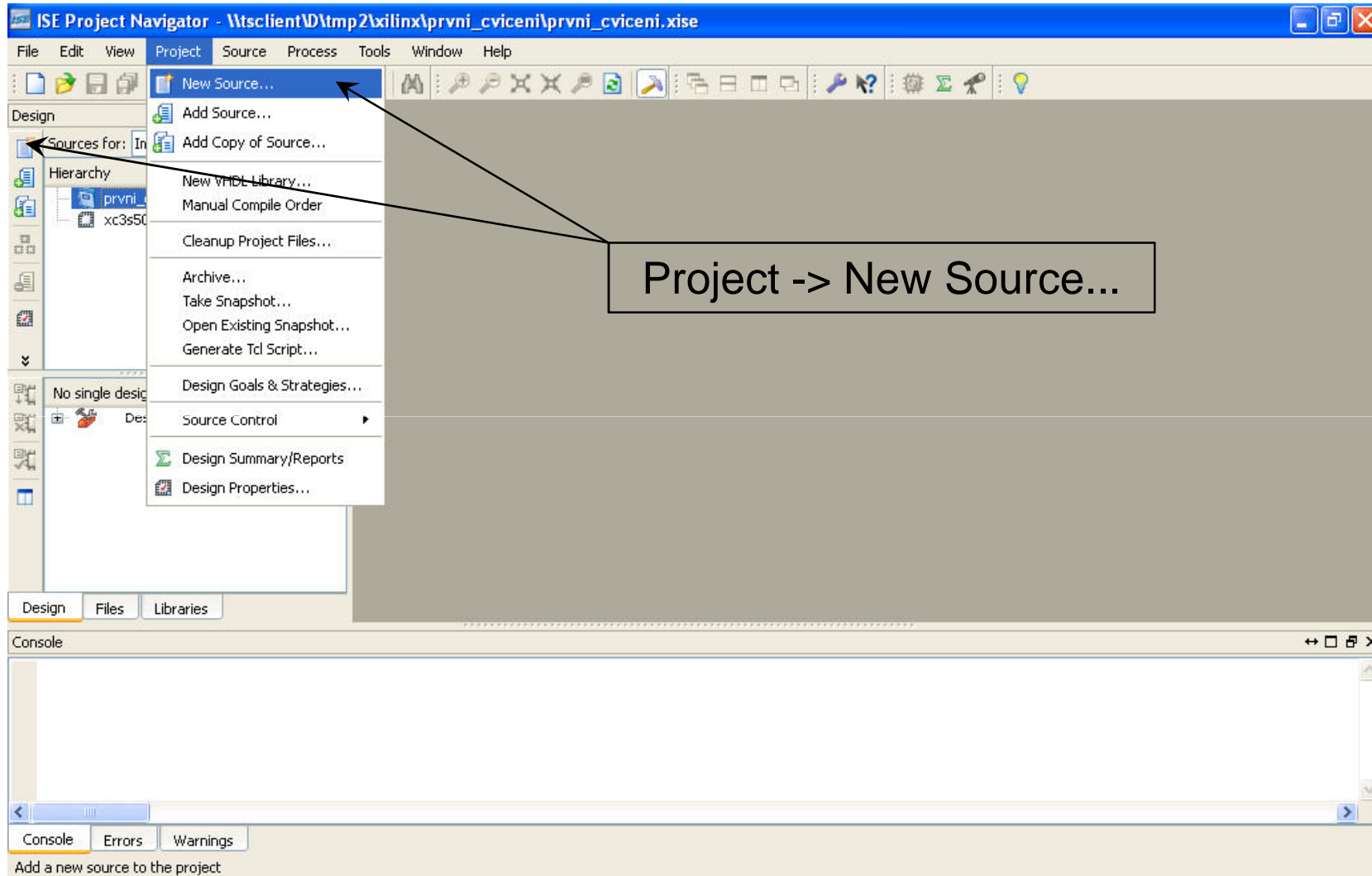
# Založení nového projektu v prostředí Xilinx IDE



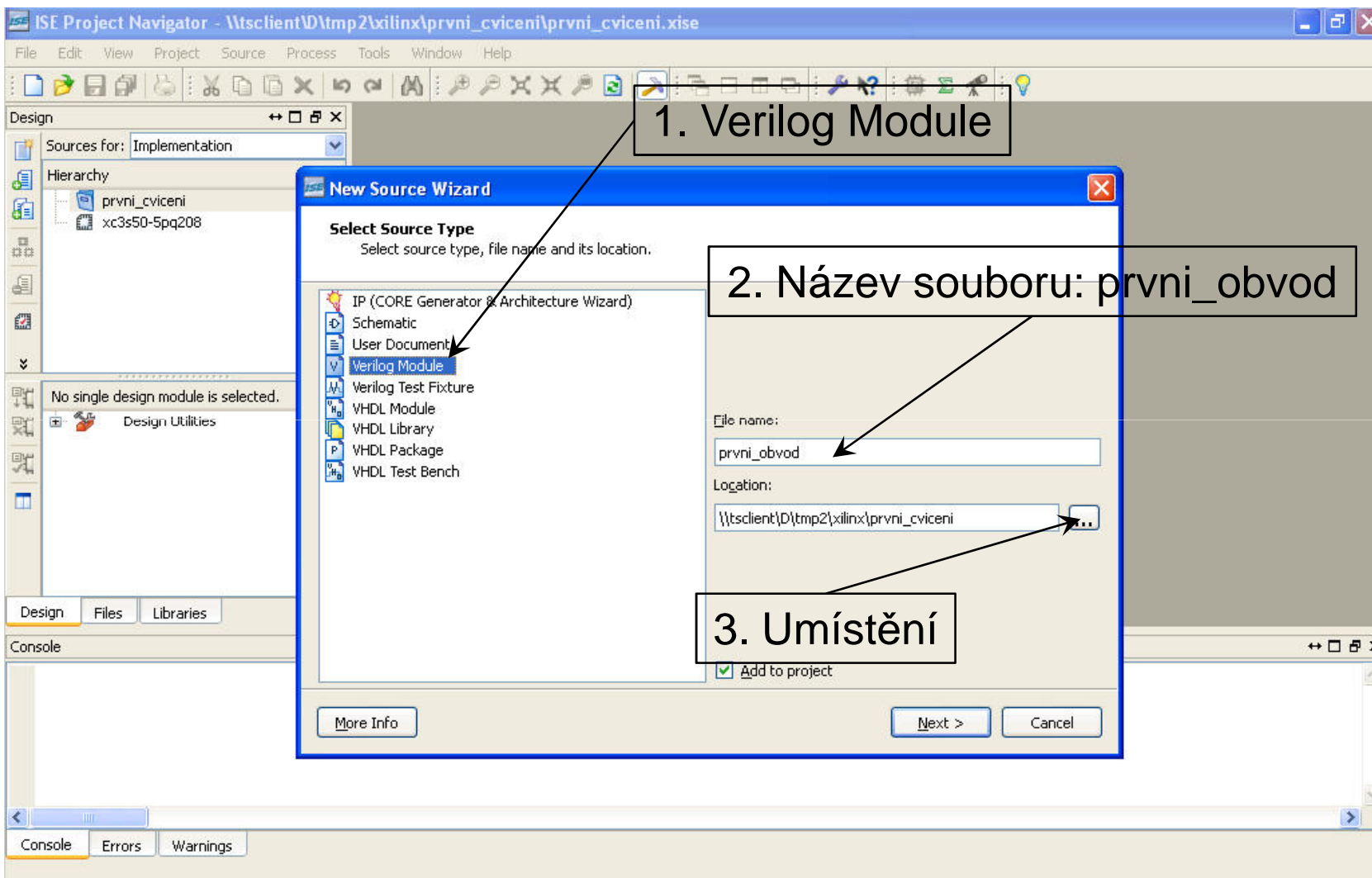
# Založení nového projektu v prostředí Xilinx IDE



# Popis tohoto obvodu v prostředí Xilinx IDE



# Popis tohoto obvodu v prostředí Xilinx IDE



# Popis tohoto obvodu v prostředí Xilinx IDE

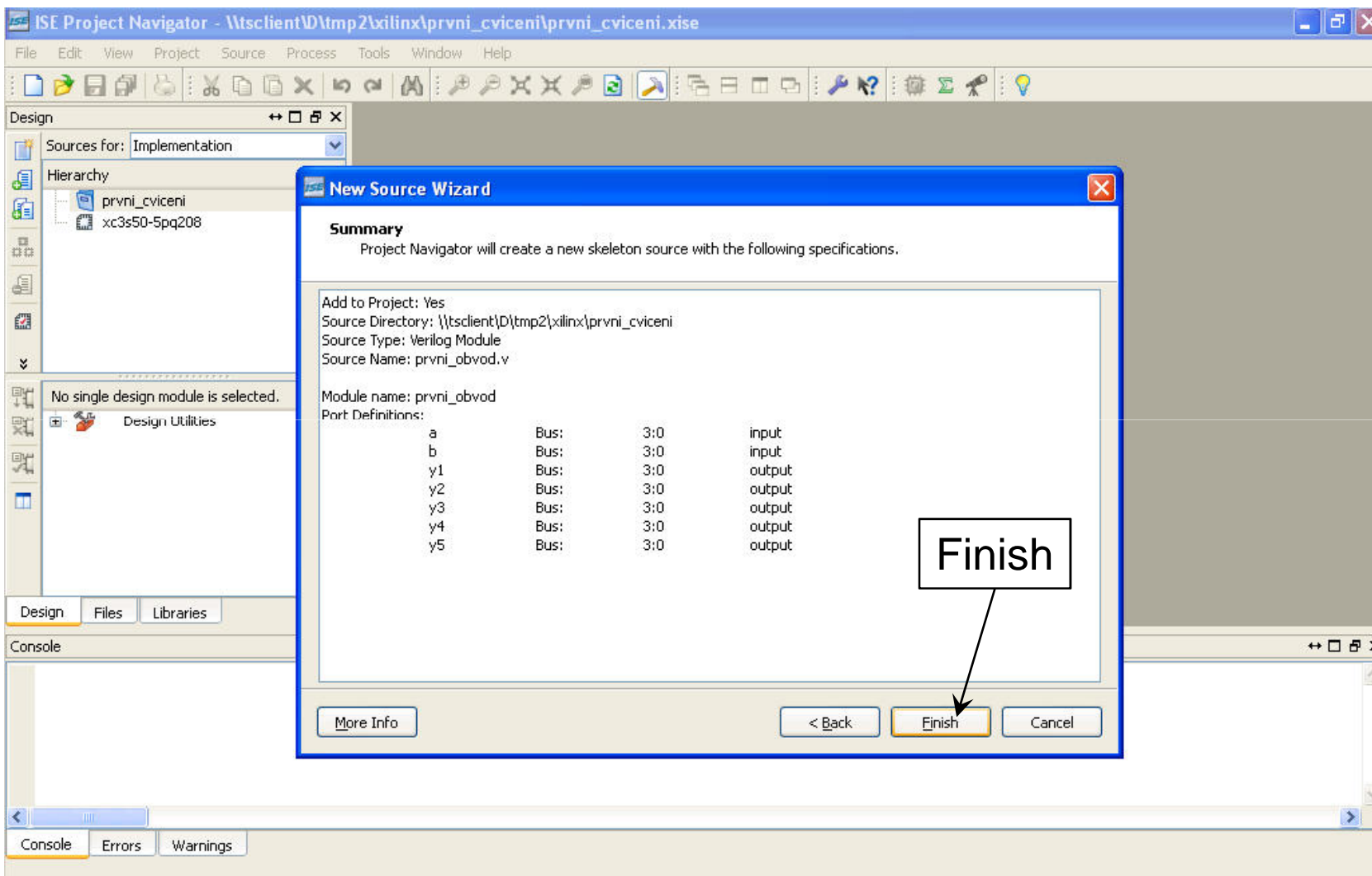
The screenshot shows the Xilinx IDE interface with the 'New Source Wizard' dialog open. The dialog is titled 'Define Module' and prompts the user to 'Specify ports for module.' The module name is 'prvni\_obvod'. The port configuration table is as follows:

Port Name	Direction	Bus	MSB	LSB
a	input	<input checked="" type="checkbox"/>	3	0
b	input	<input checked="" type="checkbox"/>	3	0
y1	output	<input checked="" type="checkbox"/>	3	0
y2	output	<input checked="" type="checkbox"/>	3	0
y3	output	<input checked="" type="checkbox"/>	3	0
y4	output	<input checked="" type="checkbox"/>	3	0
y5	output	<input checked="" type="checkbox"/>	3	0
	input	<input type="checkbox"/>		
	input	<input type="checkbox"/>		
	input	<input type="checkbox"/>		
	input	<input type="checkbox"/>		
	input	<input type="checkbox"/>		

Annotations in the image include 'vstupy' (inputs) pointing to the 'a' and 'b' rows, and 'výstupy' (outputs) pointing to the 'y1' through 'y5' rows.

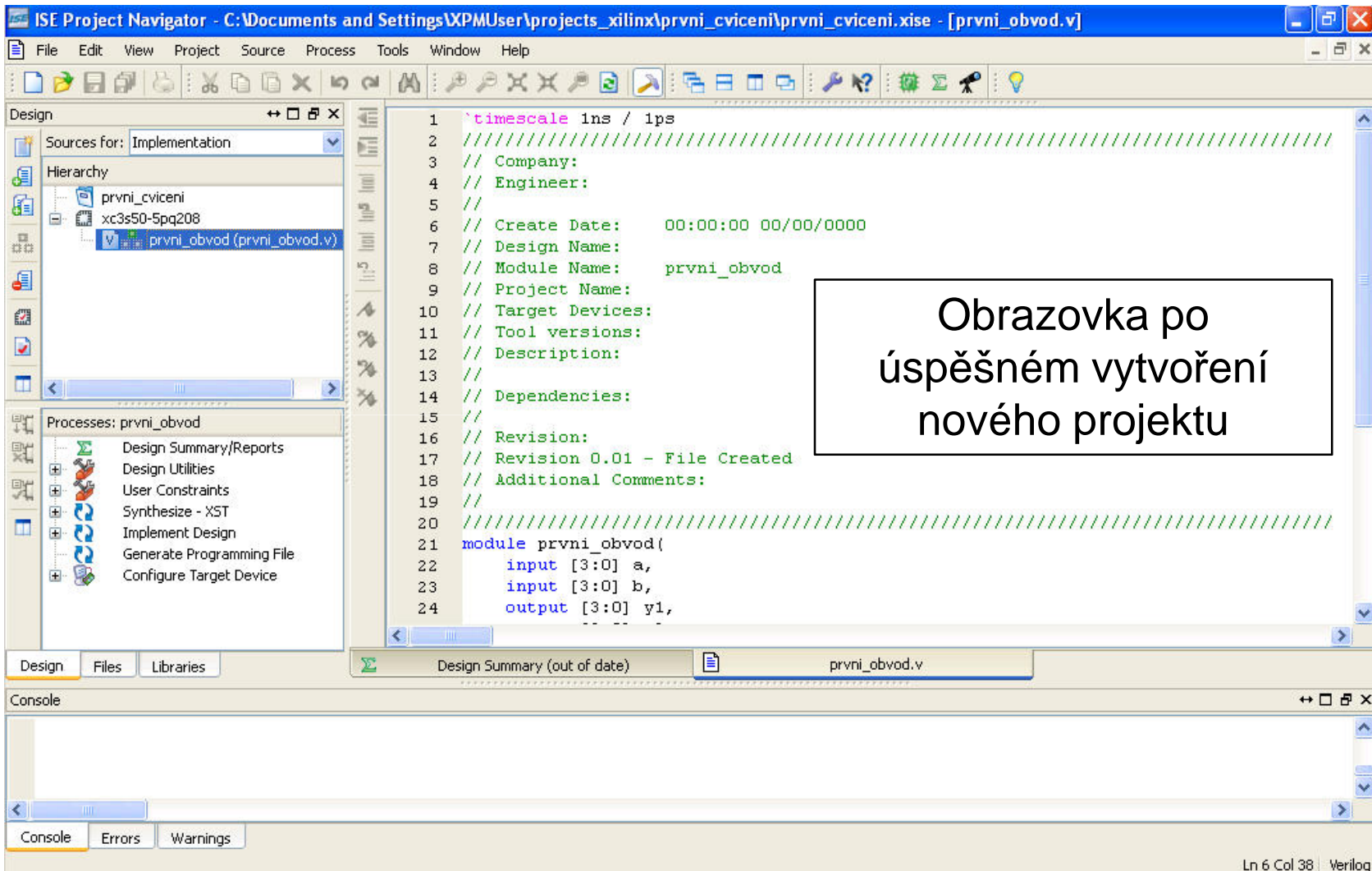


# Popis tohoto obvodu v prostředí Xilinx IDE





# Popis tohoto obvodu v prostředí Xilinx IDE



# Popis tohoto obvodu v prostředí Xilinx IDE

The screenshot shows the Xilinx ISE IDE interface. The main window displays a Verilog code editor with the following code:

```
15 //  
16 // Revision:  
17 // Revision 0.01 - File Created  
18 // Additional Comments:  
19 //  
20 ///////////////////////////////////////////////////////////////////  
21 module prvni_obvod(  
22     input [3:0] a,  
23     input [3:0] b,  
24     output [3:0] y1,  
25     output [3:0] y2,  
26     output [3:0] y3,  
27     output [3:0] y4,  
28     output [3:0] y5  
29 );  
30  
31     assign y1 = a & b; // AND  
32     assign y2 = a | b; // OR  
33     assign y3 = a ^ b; // XOR  
34     assign y4 = ~(a & b); // NAND  
35     assign y5 = ~(a | b); // NOR  
36  
37 endmodule  
38
```

A callout box on the right side of the code editor contains the text: "Dopsání popisu obvodu:" followed by the same code lines 31-35. An arrow points from this callout box to the corresponding code lines in the editor.

The IDE interface includes a Design window on the left showing the project hierarchy, a Processes window below it, and a Console window at the bottom showing the launch sequence.

## Verilog – simulace – Obecně

- **Krok první** – „zapouzdřit“ simulovaný obvod do modulu bez vstupů a výstupů
- **Krok druhý** – vytvořit vnitřní proměnné tohoto modulu (reg, wire) pro nastavování vstupů (reg) a sledování výstupů (wire) simulovaného obvodu
- **Krok třetí** – přiřazení vytvořených vnitřních proměnných vstupům a výstupům simulovaného obvodu
- **Krok čtvrtý** – specifikace časové posloupnosti stimulů obvodu

# Verilog – simulace – Obecně

```
`timescale 1ns / 1ps
```

```
module test();
```

žádné vstupy ani výstupy

```
reg a, b; // vstupy  
wire c; // výstupy
```

vytvoření vnitřních proměnných

```
nazev_simulovaneho_obvodu pojmenovani_instance(  
.a(a),  
.b(b),  
.c(c),  
);
```

Signály s tečkou před jménem jsou názvy signálů uvnitř simulovaného obvodu, zatímco v závorkách jsou názvy vnitřních proměnných

vytvoření instance simulovaného obvodu a přiřazení vnitřních proměnných vstupům a výstupům simulovaného obvodu

```
initial begin  
a = 0;  
b = 0;  
end
```

inicializace vstupů

```
always #40 a = ~a;
```

```
always #80 b = ~b;
```

Všechny always bloky (více příkazů pro jeden always se ohraničuje mezi begin a end) jsou vykonávány současne...

```
endmodule
```

invertuj b každých 80 ns

# Simulace tohoto obvodu v prostředí Xilinx IDE

ISE Project Navigator - \\tsclient\D\tmp2\xilinx\prvni\_cviceni\prvni\_cviceni.xise - [prvni\_obvod.v\*]

File Edit View Project Source Process Tools Window Help

Design 15 // 16 // Revision: 17 // Revision: 0.01 - File Created

1. Project -> New Source, nebo použít ikonu..

2. Verilog Test Fixture

3. Název souboru pro definici stimulů pro obvod

New Source Wizard

Select Source Type  
Select source type, file name and its location.

- BMM File
- IP (CORE Generator & Architecture Wizard)
- Implementation Constraints File
- MEM File
- Schematic
- User Document
- Verilog Module
- Verilog Test Fixture
- Verilog Test Bench
- WHDL Module
- WHDL Library
- WHDL Package
- WHDL Test Bench

File name: test

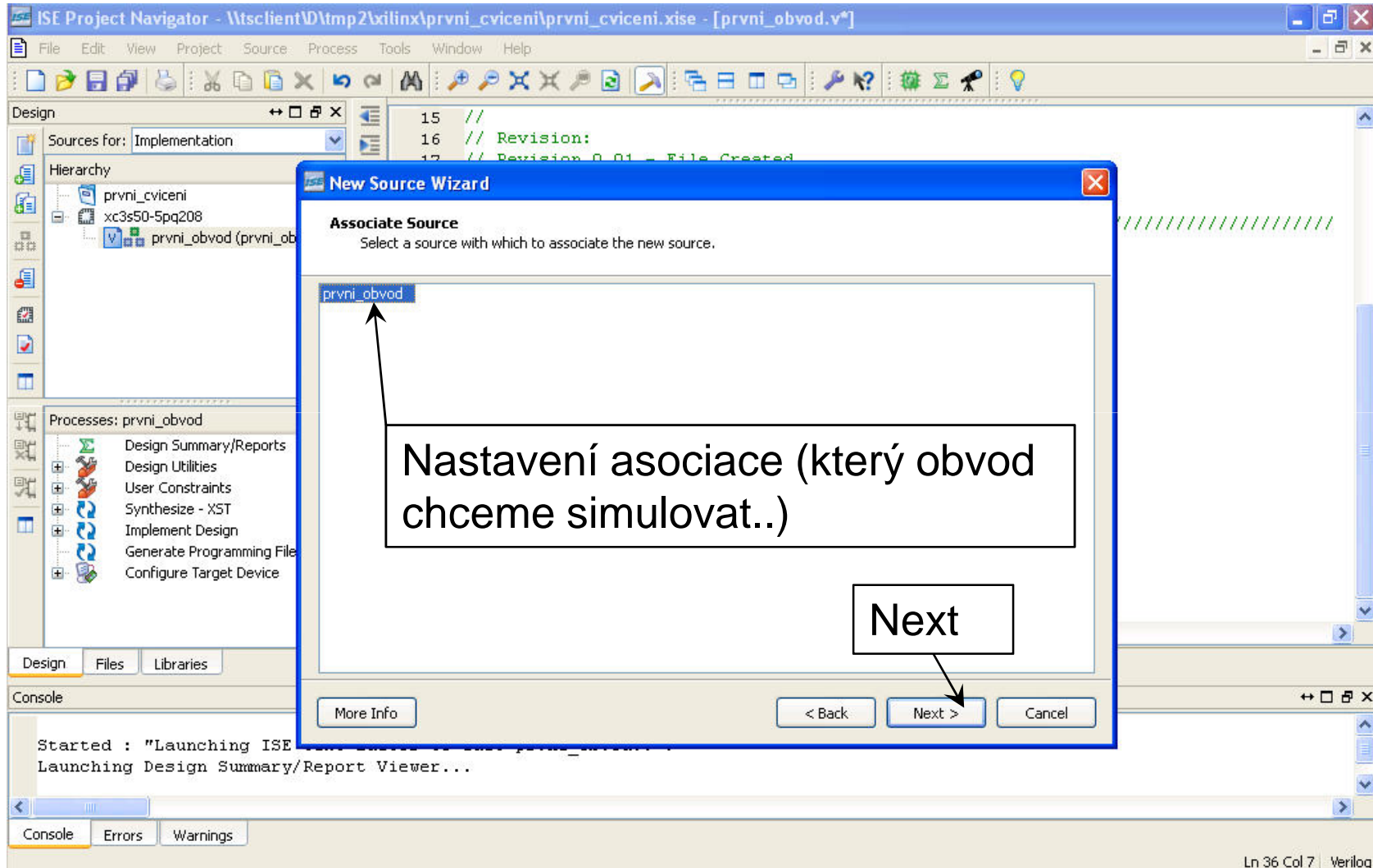
Location: \\tsclient\D\tmp2\xilinx\prvni\_cviceni

More Info Next > Cancel

Started : "Launching ISE  
Launching Design Summary/Report Viewer..."

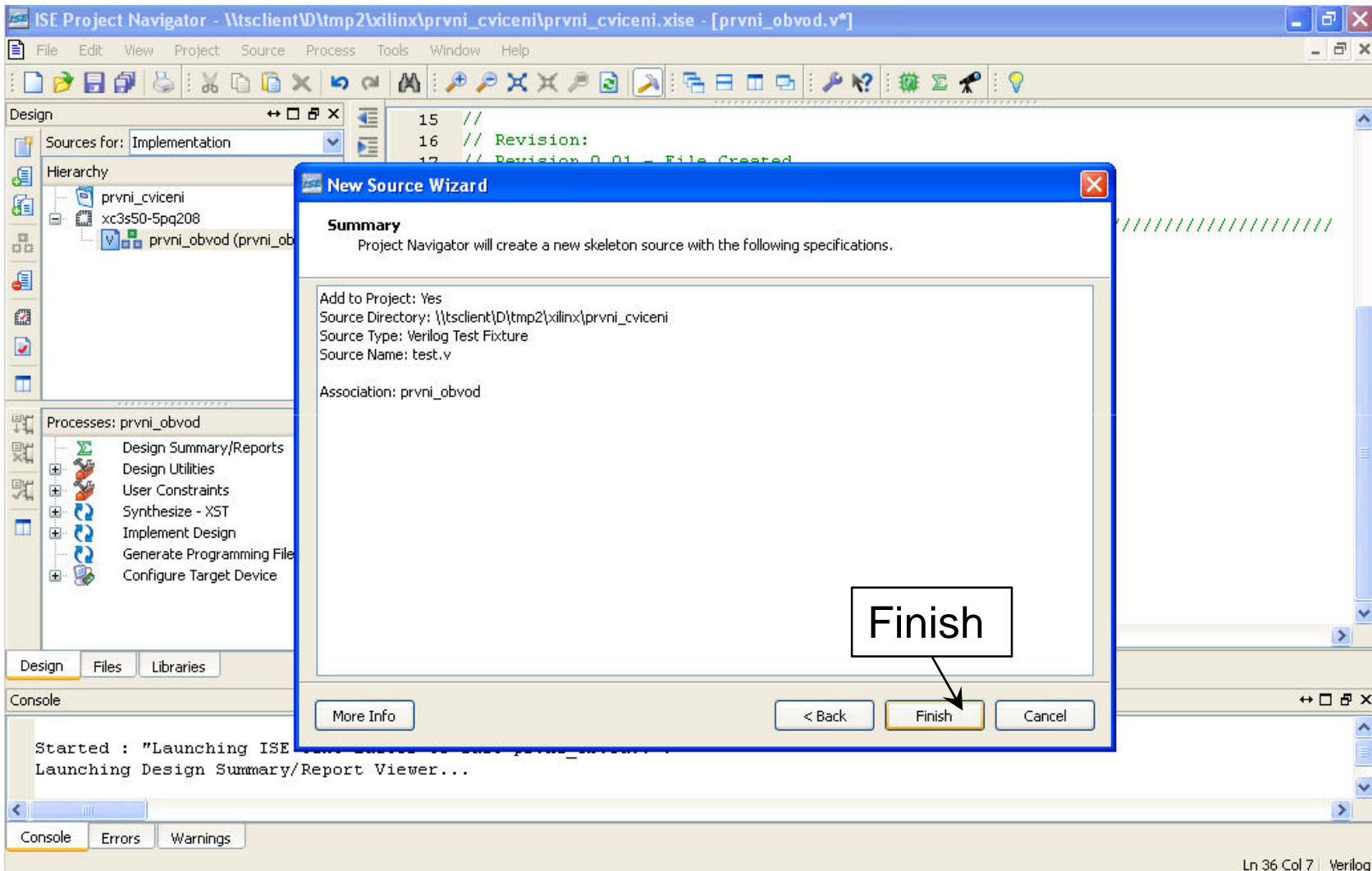
Ln 36 Col 7 Verilog

# Simulace tohoto obvodu v prostředí Xilinx IDE





# Simulace tohoto obvodu v prostředí Xilinx IDE





# Simulace tohoto obvodu v prostředí Xilinx IDE

**Předpřipravené.. (Netřeba měnit)**

**Dopsání vnějších stimulů obvodu (lze to i lépe.. Viz str.17):**

```
a = 4'b0000; // binárně  
b = 4'b1111;  
  
#100; // pozastavení na 100 jednotek  
a = 15; // dekadicky  
b = 0;  
  
#100;  
a = 4'hF; // hexadecimálně  
b = 'hF;
```

Started : "Launching ISE Text Editor to edit test.v".

# Simulace tohoto obvodu v prostředí Xilinx IDE

The screenshot shows the Xilinx ISE IDE interface. The Design window on the left displays a Hierarchy tree with the following structure:

- Sources for: Behavioral Simulation
- Hierarchy
  - prvni\_cviceni
    - xc3s50-5pq208
      - test (test.v)
      - uut - prvni\_obvod (prvni\_obv.v)

The Processes window shows the following steps:

- ISim Simulator
  - Behavioral Check Syntax
  - Simulate Behavioral Model

The main code editor displays the following Verilog code:

```
47 );  
48  
49 initial begin  
50 // Initialize Inputs  
51 a = 0;  
52 b = 0;  
53  
54 // Wait 100 ns for global reset to finish  
55 #100;  
56  
57 // Add stimulus here  
58 a = 4'b0000;  
59 b = 4'b1111;  
60  
61 #100;  
62 a = 15;  
63 b = 0;  
64  
65 #100;  
66 a = 4'hF;  
67 b = 'hF;  
68  
69  
70 end
```

Three callout boxes provide instructions:

1. Výběr: Behavioral Simulation
2. Výběr souboru: test
3. Simulate Behavioral Model (Behavioral Check Syntax se provede automaticky)

The Console window at the bottom shows the message: "Started : "Launching ISE Text Editor to edit test.v"."

# Simulace tohoto obvodu v prostředí Xilinx IDE

The screenshot displays the Xilinx ISim simulation environment. The main window is titled "ISim - [Default.wcfg\*]" and contains several panes:

- Instances and Process Name:** Shows a tree view with "test" and "gbl" under "Instance and Process Name".
- Simulation Objects for test:** A table listing simulation objects and their values.
- Timing Diagram:** A waveform viewer showing digital signals over time. A vertical yellow cursor is positioned at 427,500 ps.
- Console:** Displays simulation warnings and messages.

Name	Value
y1[3:0]	1111
y2[3:0]	1111
y3[3:0]	0000
y4[3:0]	0000
y5[3:0]	0000
a[3:0]	1111
b[3:0]	1111

Console output:

```
WARNING: A WEBPACK license was found.  
WARNING: Please use Xilinx License Configuration Manager to check out a full ISim license.  
WARNING: ISim will run in Lite mode. Please refer to the ISim documentation for more information on the differences between the Lite and the Full version.  
This is a Lite version of ISim.  
Time resolution is 1 ps  
Simulator is doing circuit initialization process.  
Finished circuit initialization process.  
ISim>
```

Timing Diagram: Shows signals y1, y2, y3, y4, y5, a, and b. A vertical cursor is at 427,500 ps. The diagram shows transitions in signal values over time.

Sim Time : 1000 ns

Výsledek simulace...