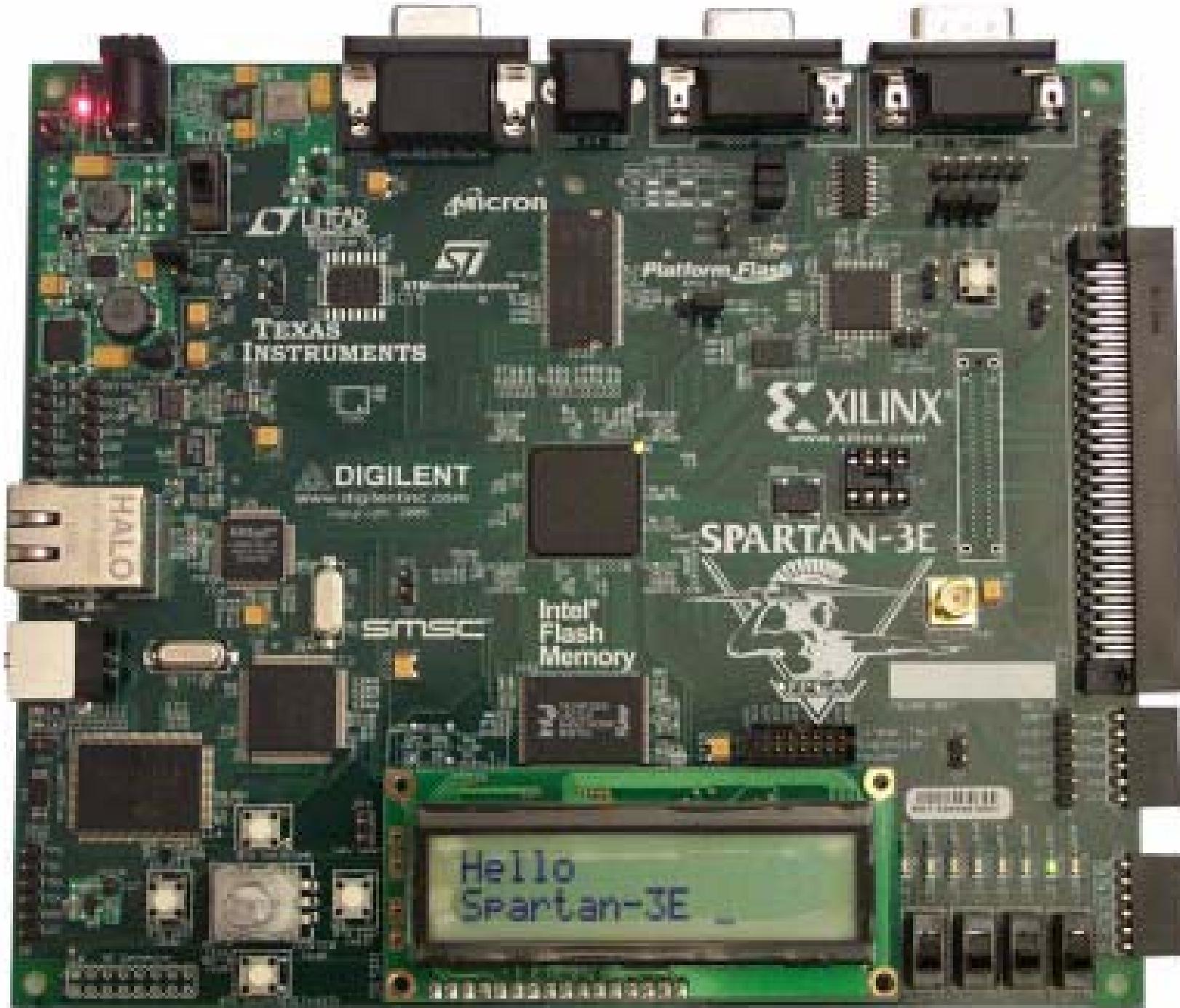
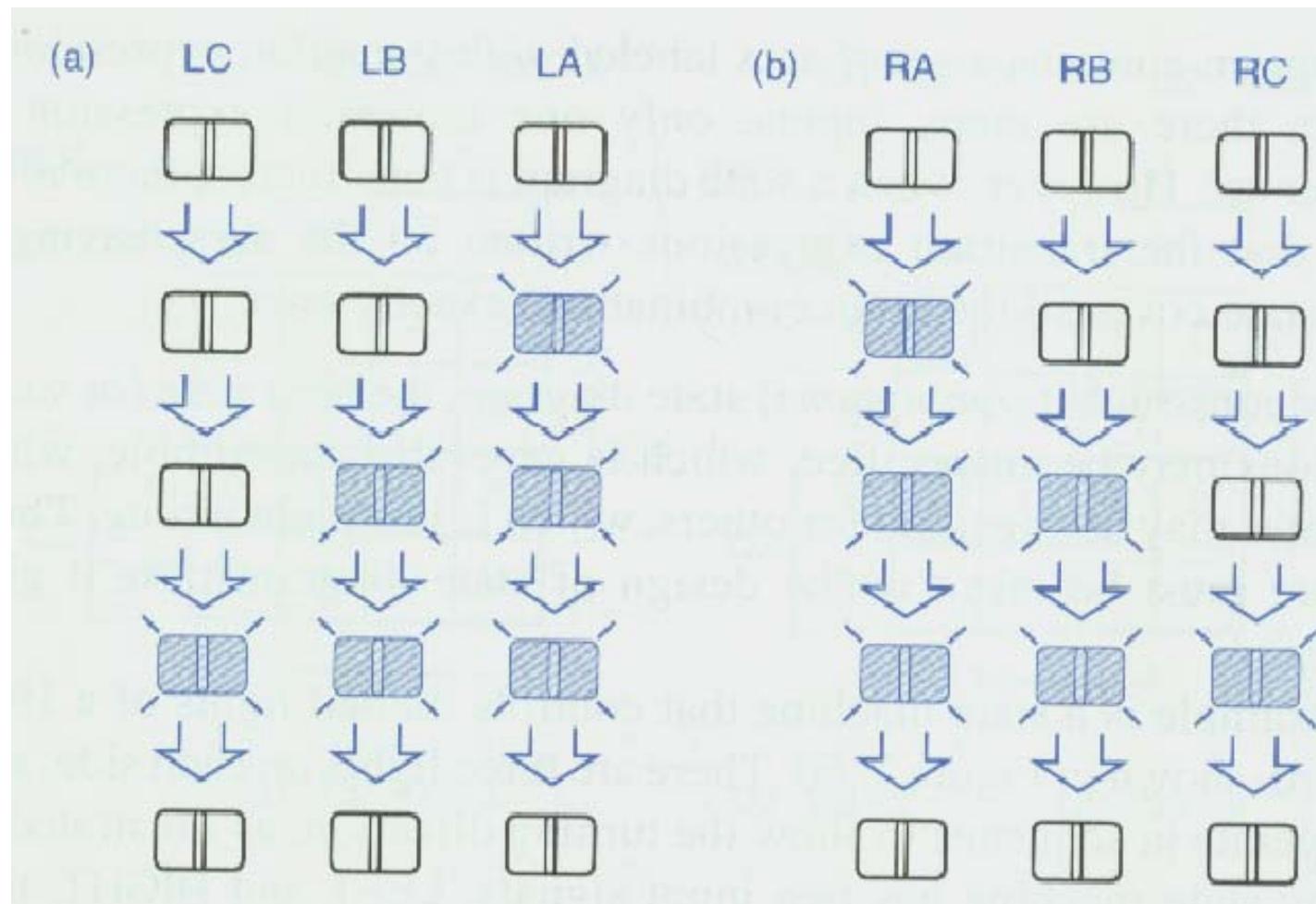


How to program the FPGA SPARTAN-3E Board



The goal of the project is to write vhdl code to control the tail lights of a 1965 Ford Thunderbird and then load this code into Spartan-3E FPGA Xilinx Board.



FPGA

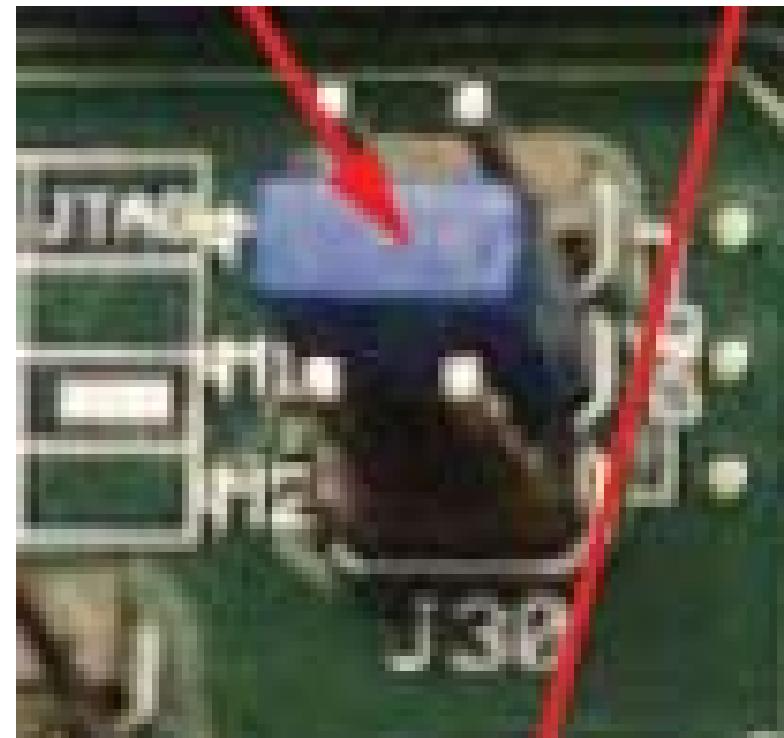
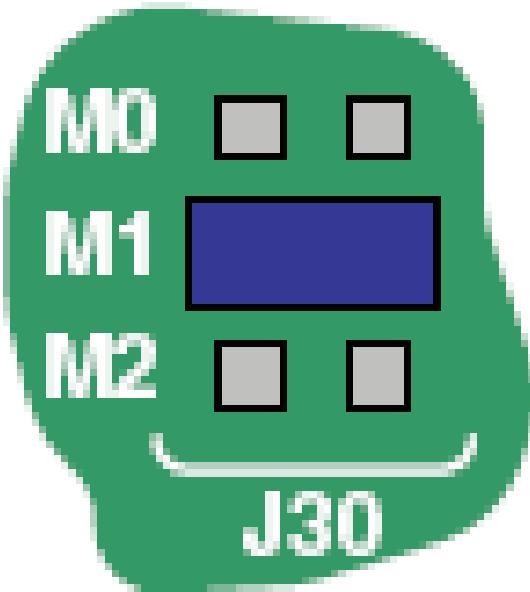
- To have a project ready to program the Spartan 3E FPGA, it was necessary to take a few preliminary steps. First, a project needed to be created in Xilinx. The project was named tail_light.
- the board has a 50MHz oscillator (see reference manual); therefore, the clock signal is so fast that the human eye is unable to see the sequence of the flashing lights. As a result, we had to add an additional file called counter.vhd, which will slow the incoming clock down.

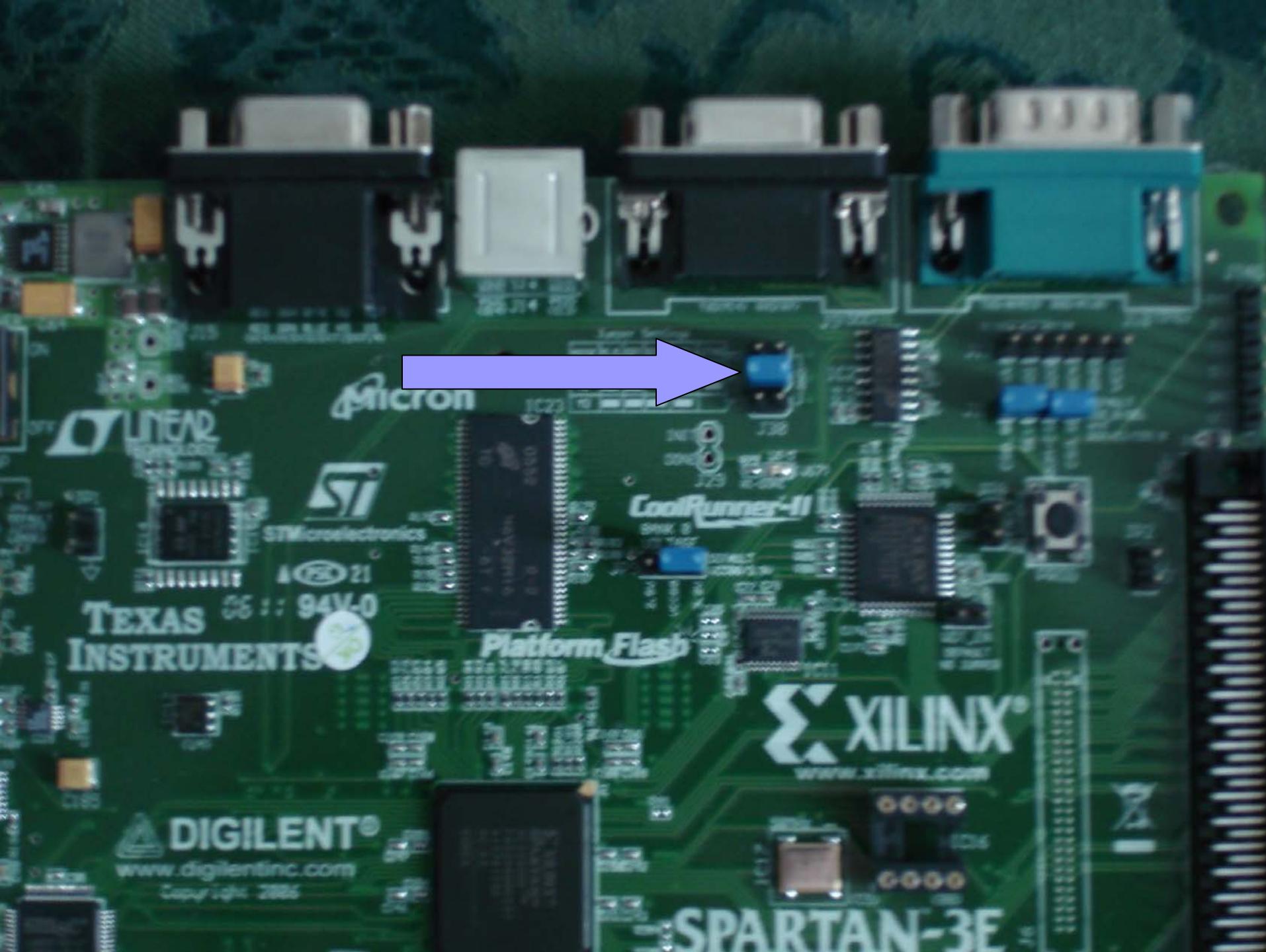
FPGA

- In order for these two files to work together, we had to write the `top_level.vhd` file

Spartan-3E Configuration Mode Jumper Settings

- Configuration Mode - JTAG
- Download from host via USB JTAG port





Platform Flash

XILINX
www.xilinx.com

DIGILENT®
www.digilentinc.com
Copyright 2005

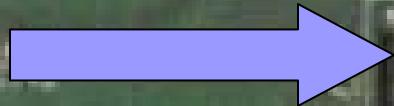
SPARTAN-3E

FPGA CLOCK INPUTS

- Clock Input FPGA Pin Global Buffer Associated DCM
- CLK_50MHZ C9 GCLK10 DCM_X0Y1
- CLK_AUX B8 GCLK8 DCM_X0Y1
- CLK_SMA A10 GCLK7 DCM_X1Y1

Platform Flash

On-Board 50 MHz Oscillator
CLK_50MHz: (C9)



SPARTAN-3E



Double click on the highlighted icon shown below. This will open the Xilinx Project Navigator window.

File Edit View Project Source Process Window Help



Sources

Sources for: Synthesis/Impl...

- tail_light
- xa2c***

Sources Snapshots

Processes

Processes:

- Add Existing Source
- Create New Source
- Design Utilities

Processes

Transcript

- Console
- Errors
- Warnings
- Find in Files

Create a new workspace by clicking File -- New Project. The New Project will be called tail_light.

start

Tutorial - Microsoft W...

Microsoft PowerPoint ...

Xilinx - ISE - Settings\...

CAPS NUM SCRLL



8:16 PM

File Edit View Project Source Process Window Help



Sources x

Sources for: Synthesis/Implementation

- tail_light
- xa2c.xco

Sources Snapshots < >

Processes x

Processes:

- Add Existing Source
- Create New Source
- Design Utilities

< >

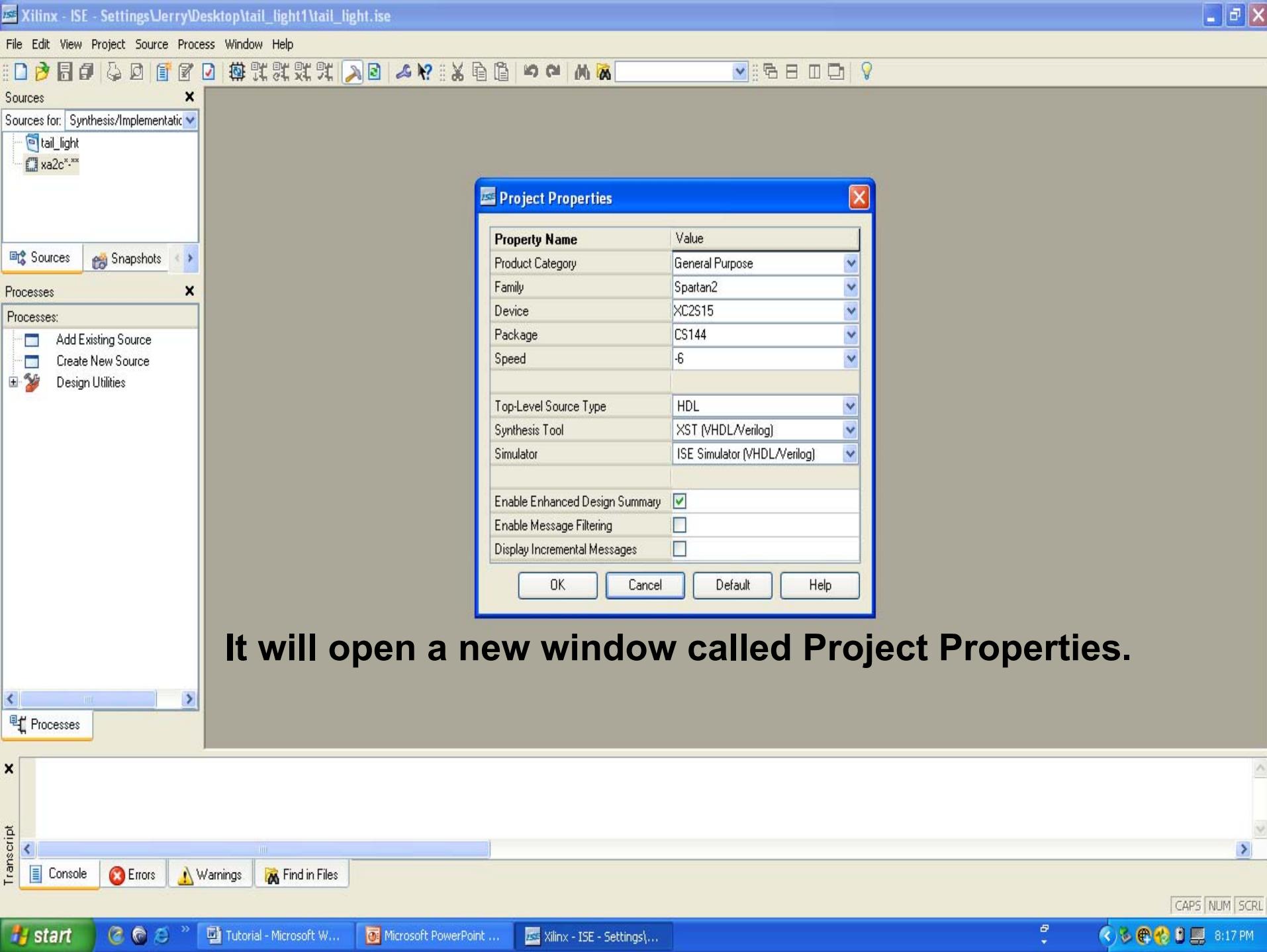
Processes

x Transcript

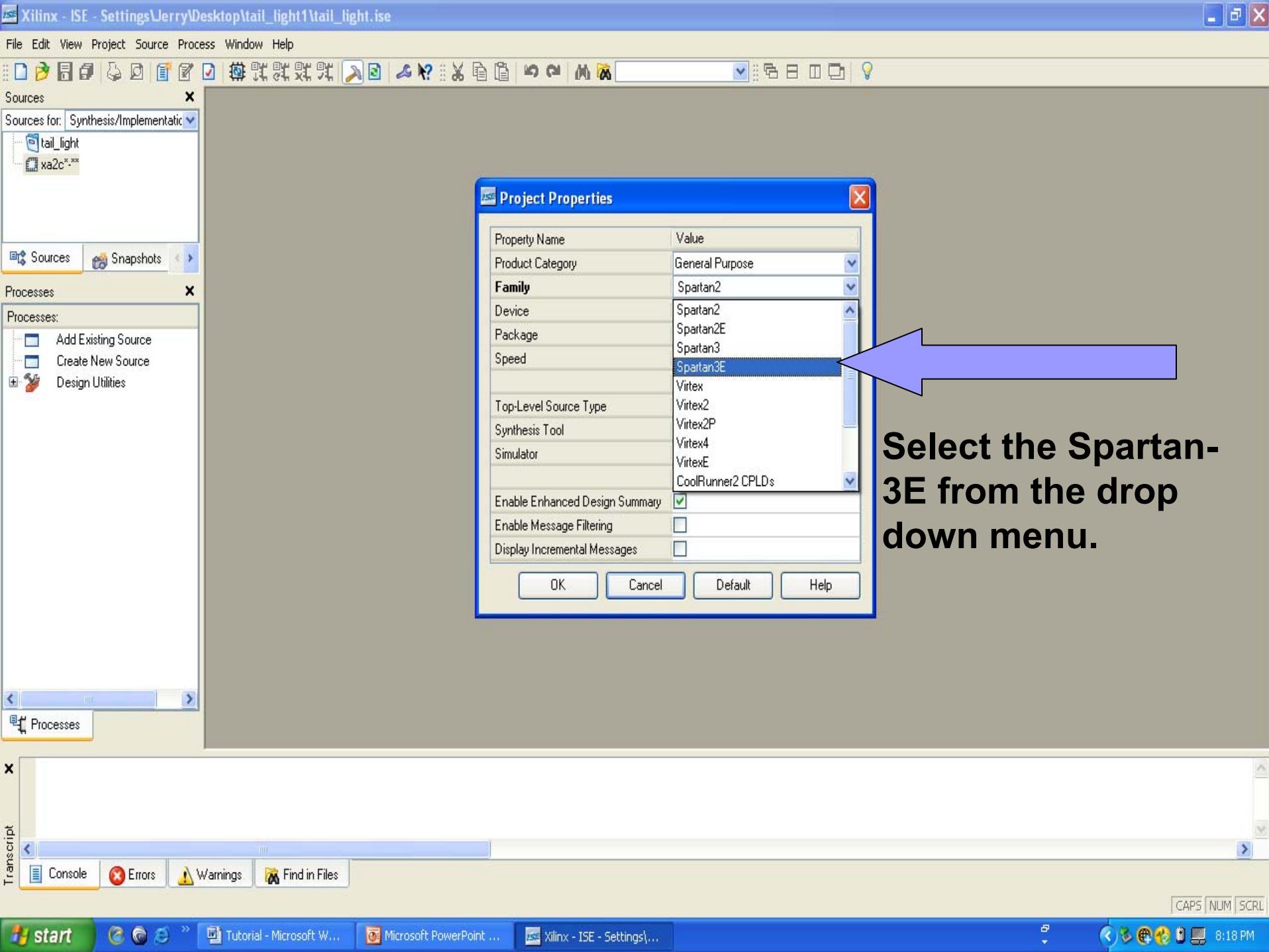
Console Errors Warnings Find in Files

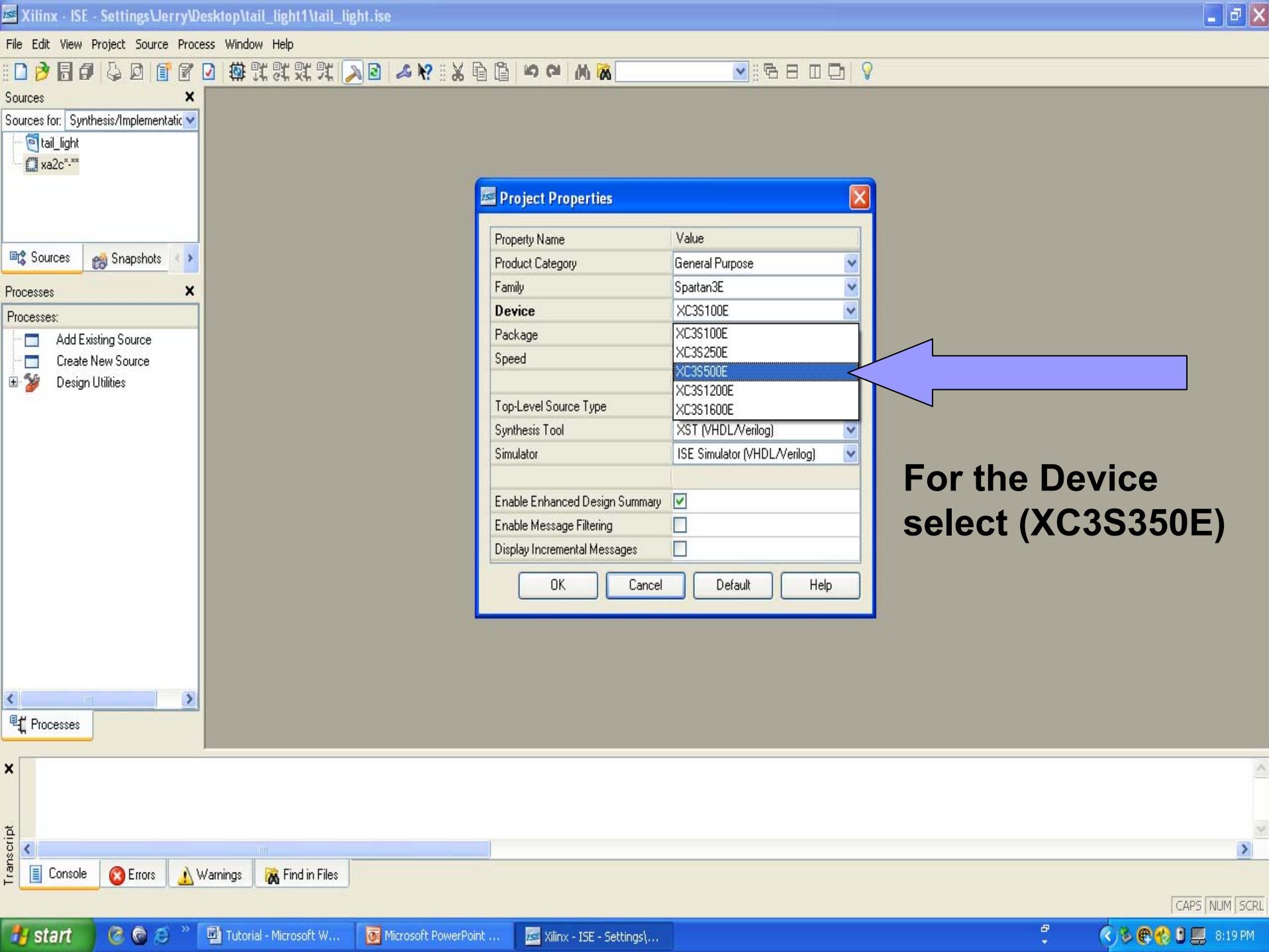
After the New Project was created, it is time to prepare the Project Properties. Double click on highlighted icon.



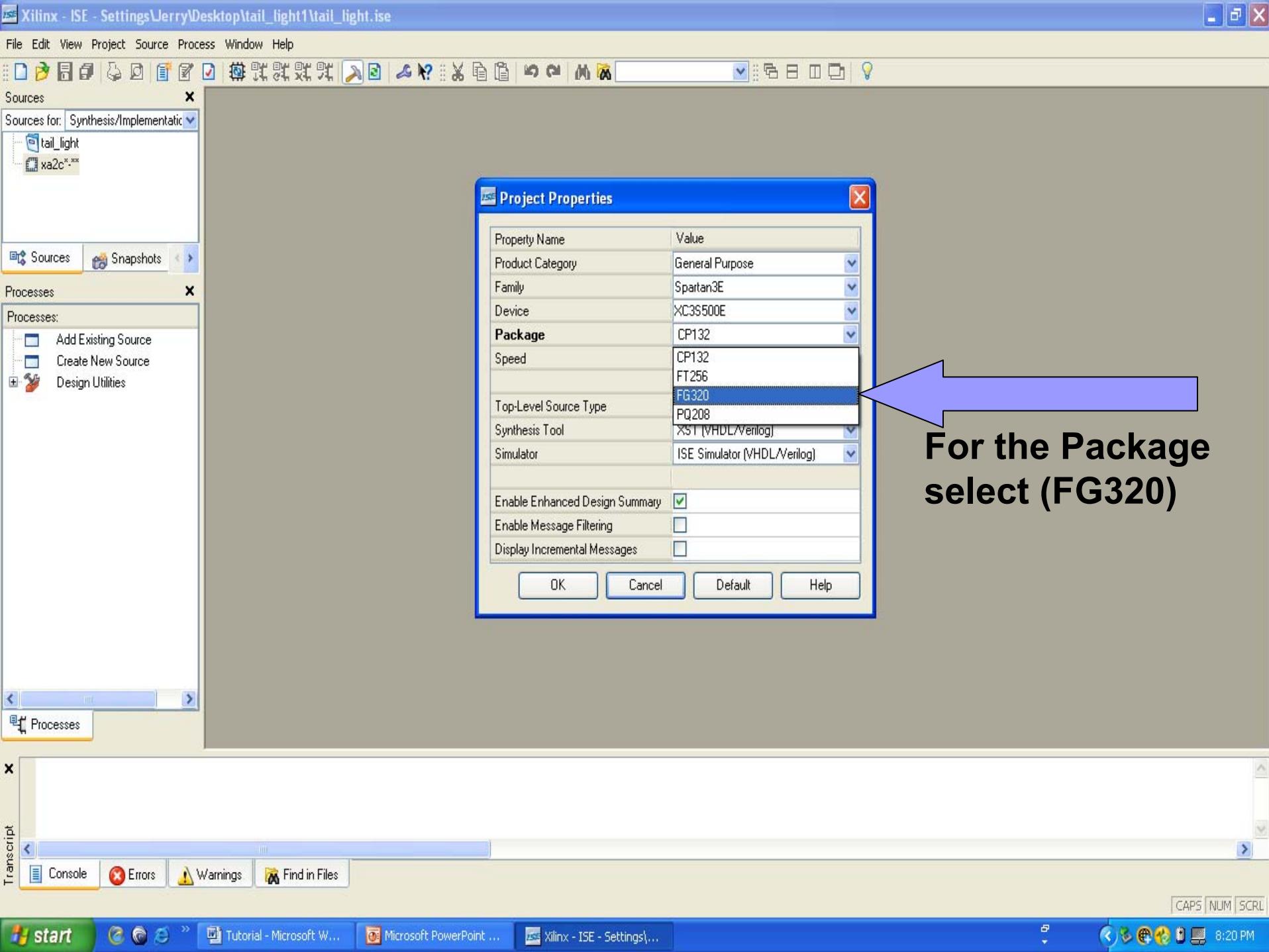


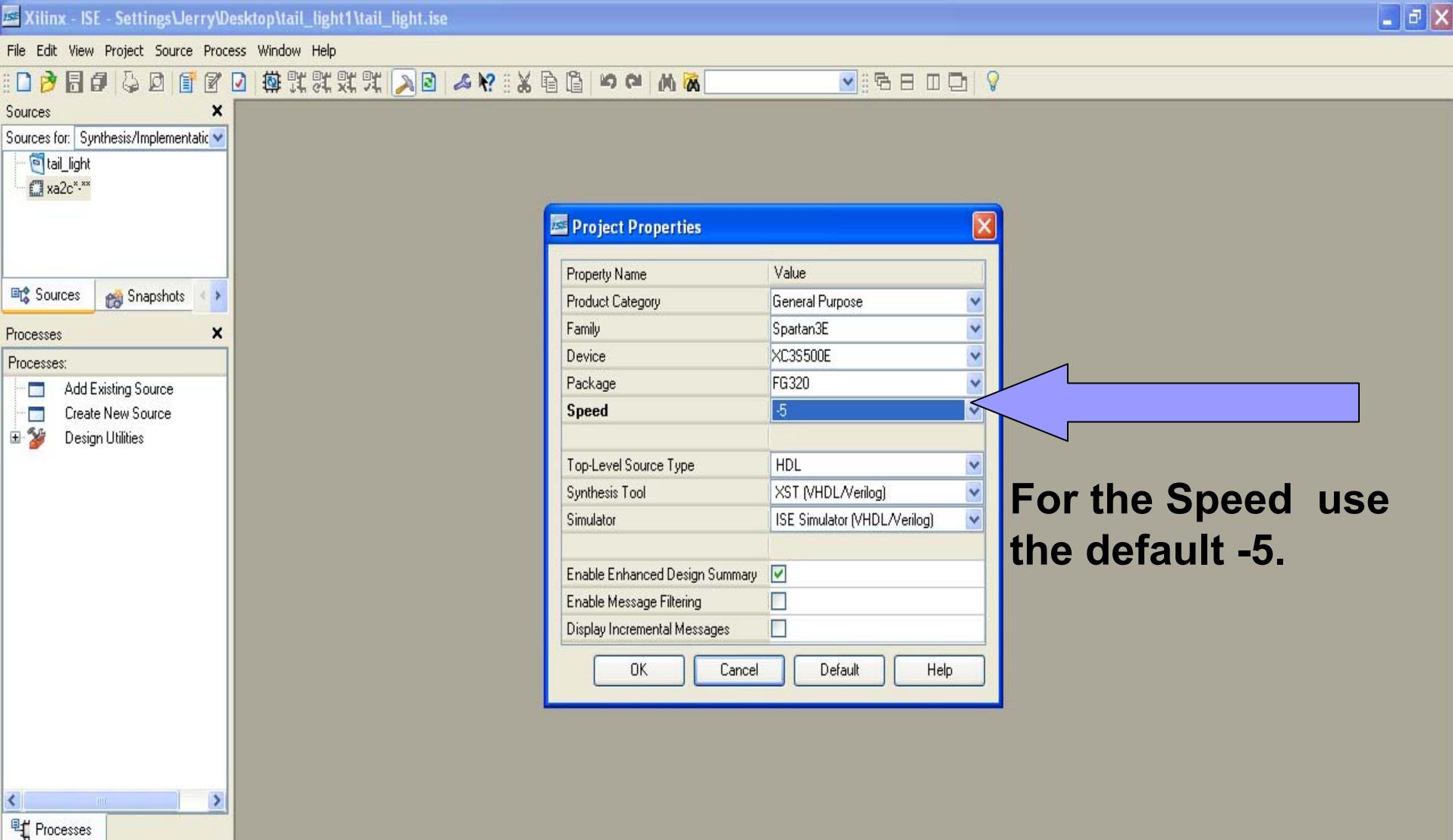
It will open a new window called Project Properties.



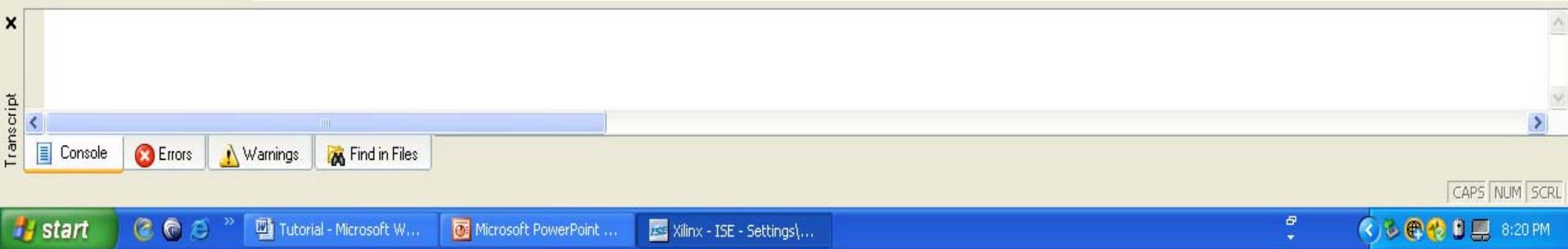


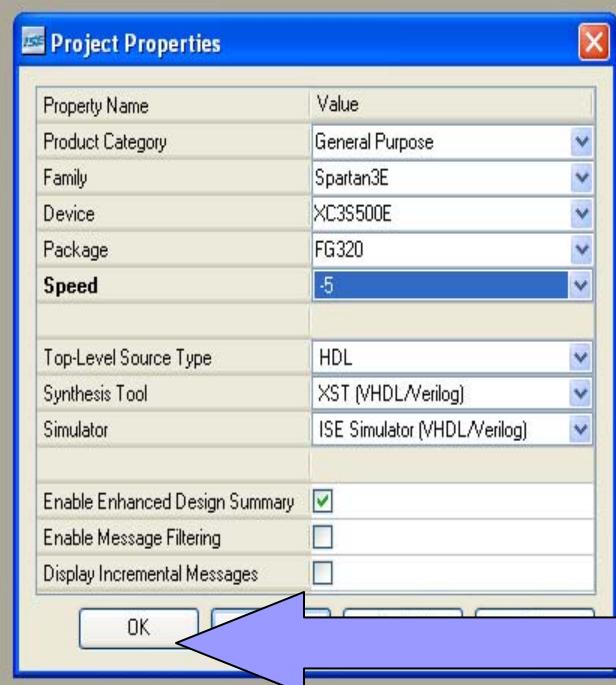
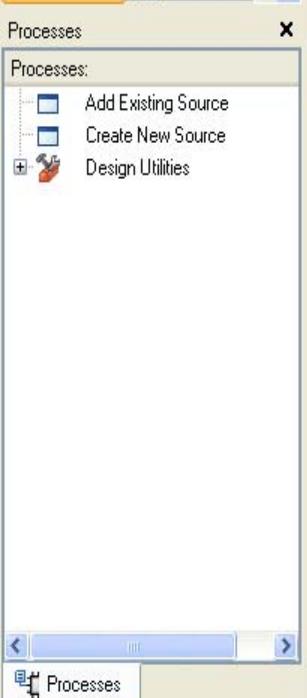
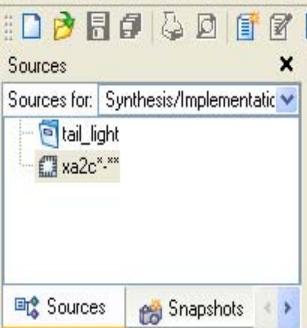
For the Device
select (XC3S350E)



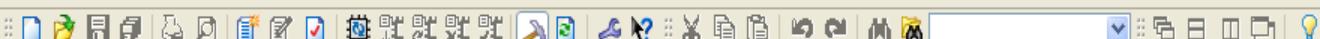


For the Speed use the default -5.





File Edit View Project Source Process Window Help

Sources x

Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320

Sources x Snapshots < >Processes x

Processes:

- Add Existing Source
- Create New Source
- Design Utilities

Processes < >Transcript xConsole xErrors xWarnings xFind in Files x

**As you can see, the Device changed to xc3s350;
otherwise, the user will not be able to assign pins.**

File Edit View Project Source Process Window Help



Sources x

Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320

Sources Snapshots < >

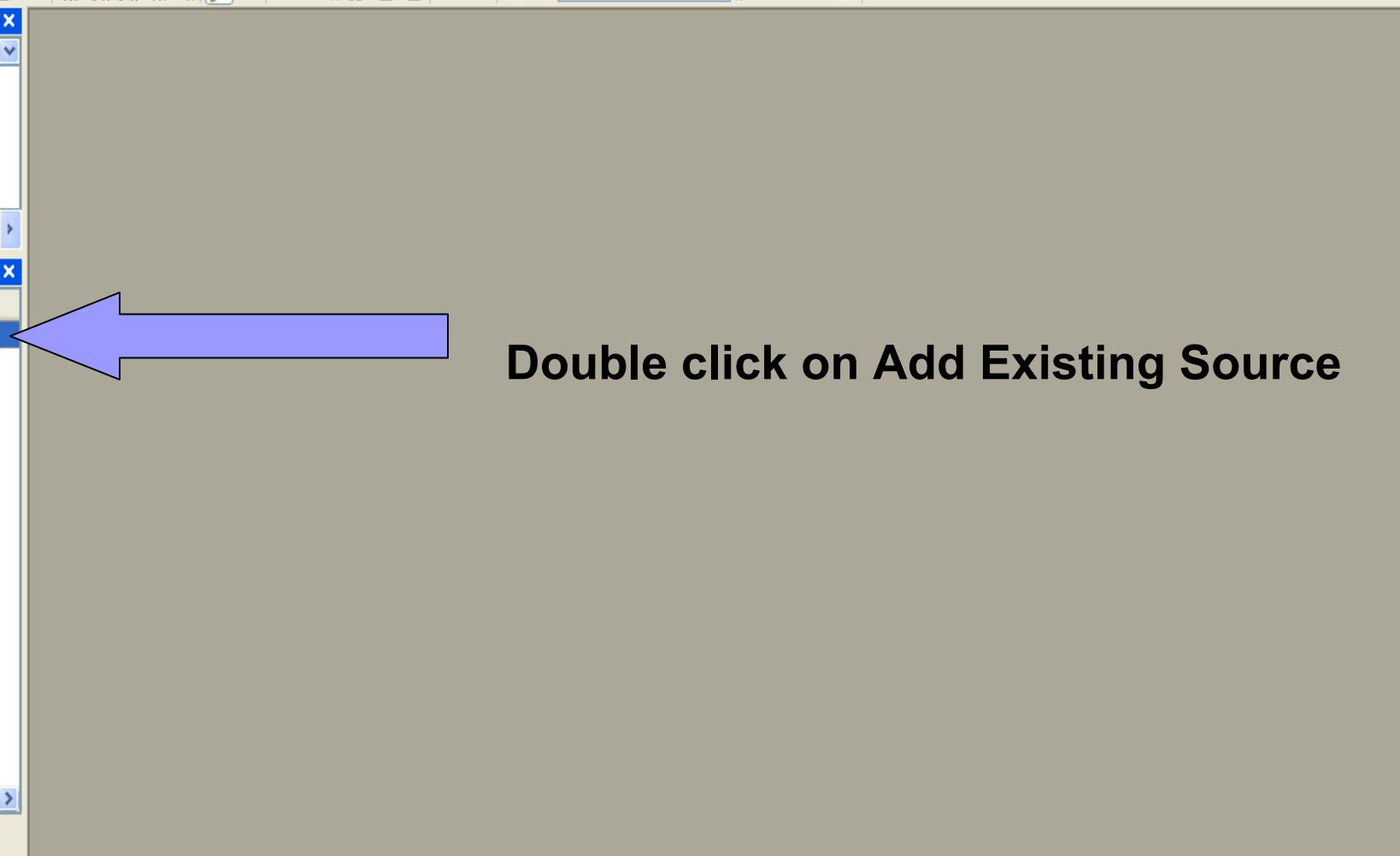
Processes x

Processes:

- Add Existing Source
- Create New Source
- Design Utilities

< >

Processes

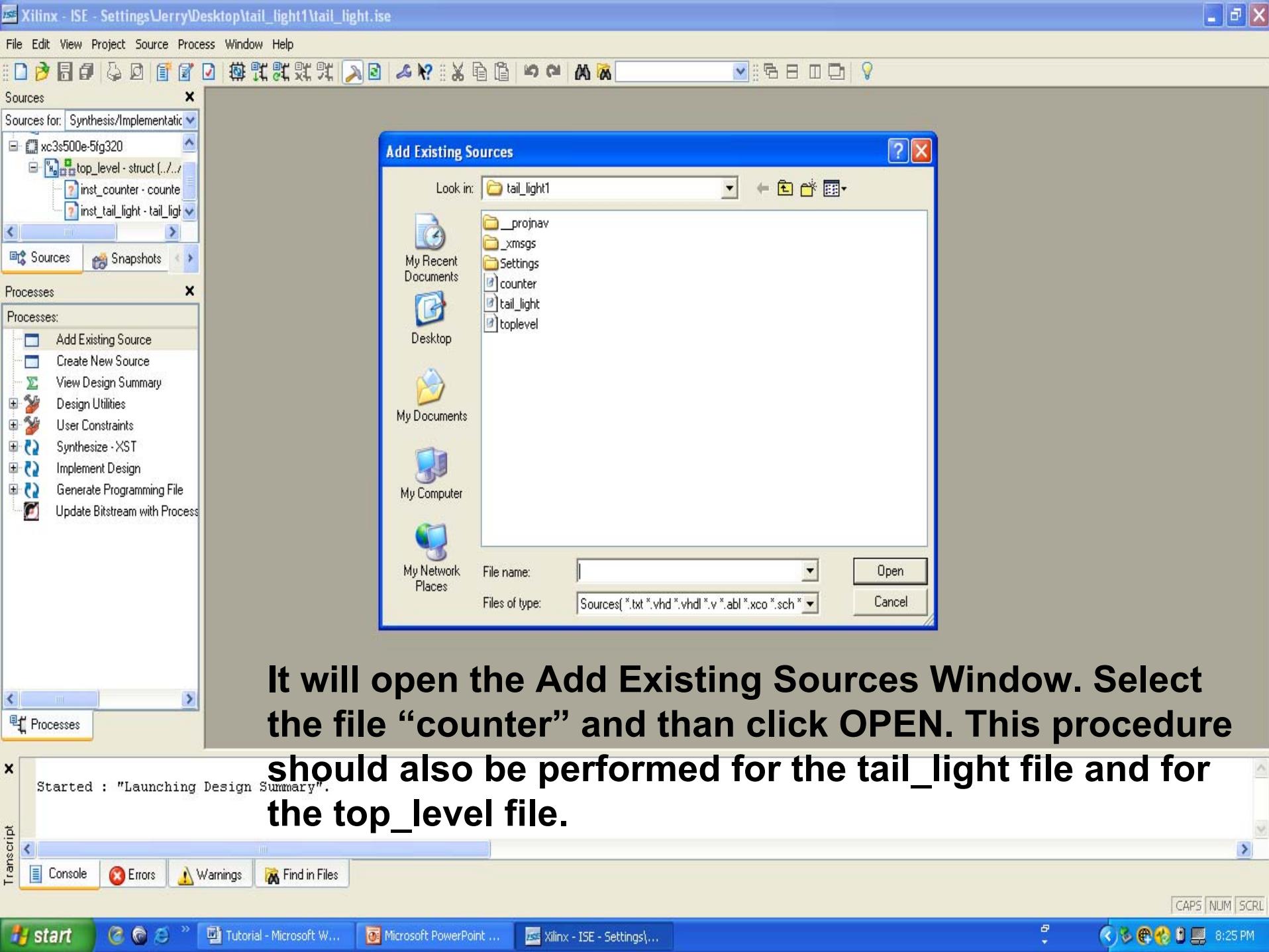


x

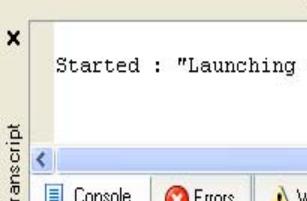
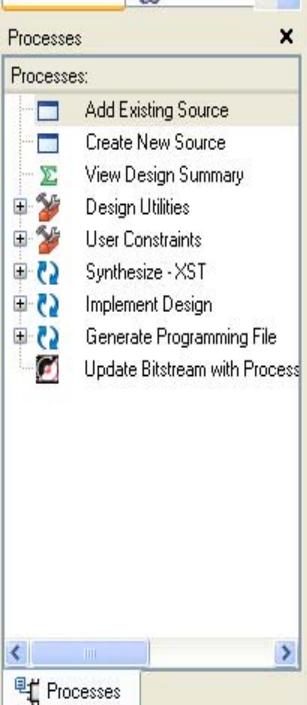
Transcript

Console Errors Warnings Find in Files

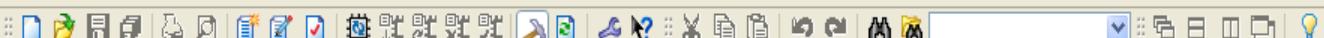
CAPS NUM SCRLL



It will open the Add Existing Sources Window. Select the file “counter” and than click OPEN. This procedure should also be performed for the tail_light file and for the top_level file.



File Edit View Project Source Process Window Help

Sources x

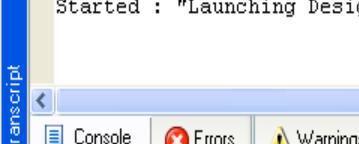
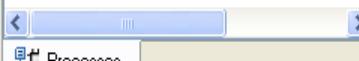
Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - struct (../../../../toplevel)
- inst_counter - counter - rtl (../../../../rtl)
- inst_tail_light - tail_light - rtl (../../../../rtl)

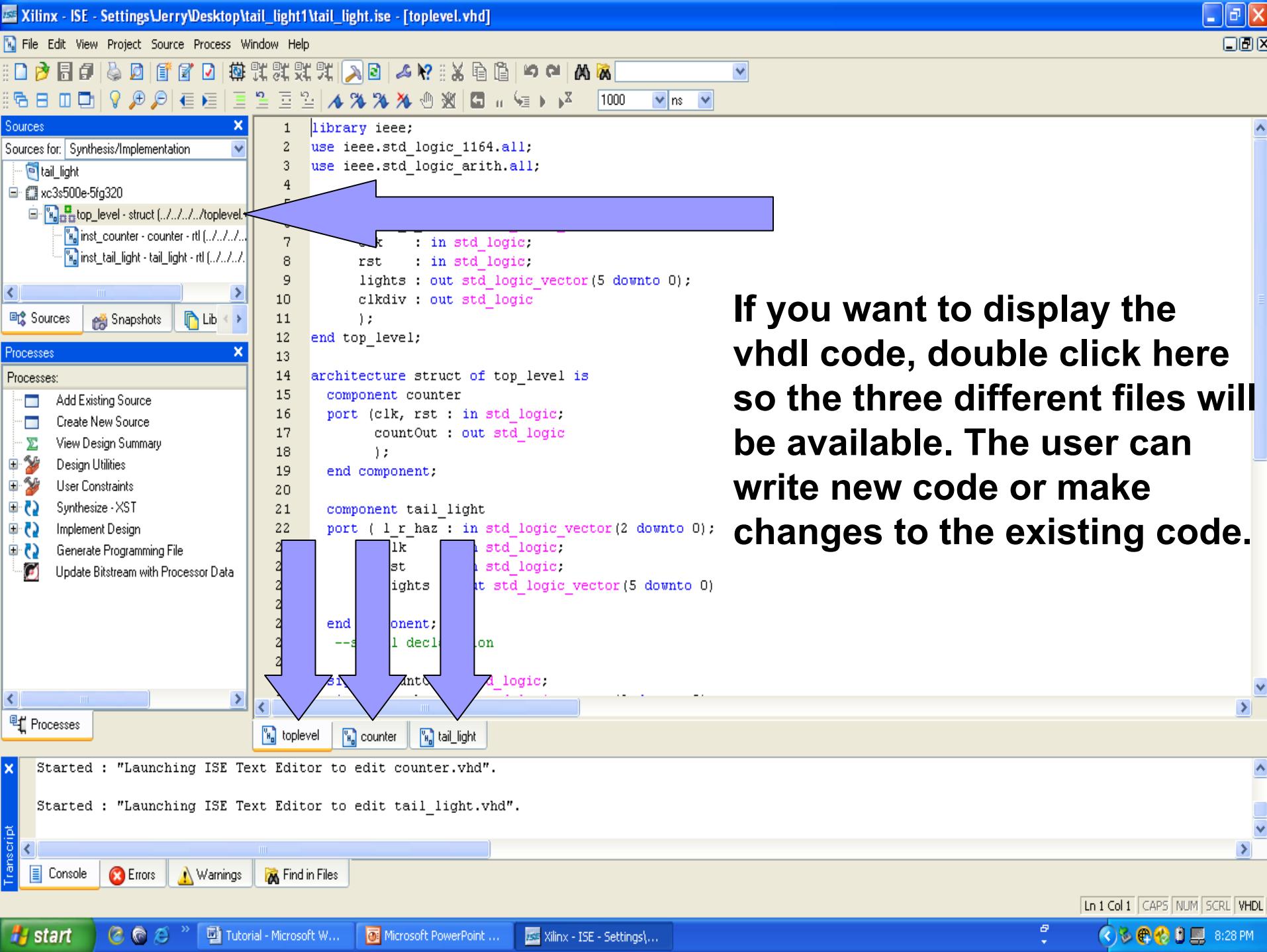
Processes x

Processes:

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Synthesize - XST
- Implement Design
- Generate Programming File
- Update Bitstream with Processor Data



As you can see we added these three files



If you want to display the vhdl code, double click here so the three different files will be available. The user can write new code or make changes to the existing code.

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - struct (../../../../toplevel)
- inst_counter - counter - rtl (../../../../counter)
- inst_tail_light - tail_light - rtl (../../../../tail_light)

Processes

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
 - Create Timing Constraints
 - Assign Package Pins
 - Create Area Constraints
 - Edit Constraints (Text)
- Synthesize - XST
- Implement Design
- Generate Programming File
- Update Bitstream with Processor Data

Started : "Launching ISE Text Editor to edit counter.vhd".

Started : "Launching ISE Text Editor to edit tail_light.vhd".

Console Errors Warnings Find in Files

1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.std_logic_arith.all;
4
5 entity top_level is
6 port(l_r_haz : in std_logic_vector(2 downto 0);
7 clk : in std_logic;
8 rst : in std_logic;
9 lights : out std_logic_vector(5 downto 0);
10 cllk : out std_logic
11);
12 end entity;
13
14 architecture struct of top_level is
15 component counter
16 port(clk, rst : in std_logic;
17 countOut : out std_logic
18);
19 end component;
20
21 Component gnt
22 port(l_r_haz : in std_logic_vector(2 downto 0);
23 clk : in std_logic;
24 rst : in std_logic;
25 lights : out std_logic_vector(5 downto 0);
26);
27 end component;
28 --signal declaration
29
30 signal countOut : std_logic;

The Processes of Source
window shows the different process that can be done to the selected code. Here the user can compile the source code, generate synthesis reports, generate a file for download into the FPGA, and download a file to an FPGA.

Expand the User Constraints in the Processes for Source window and double click Assign Package Pins. This is where you tell Xilinx which pins on the Spartan-3E will be used.

NOTE

with ISE Web pack 10.1 is called Floor Plan I/O - Presynthesis

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - struct (../../../../toplevel)
- inst_counter - counter - rtl (../../../../toplevel)
- inst_tail_light - tail_light - rtl (../../../../toplevel)

Processes

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Create Timing Constraints
- Assign Package Pins** (highlighted)
- Create Area Constraints
- Edit Constraints (Text)
- Synthesize -XST
- Implement Design
- Generate Programming File
- Update Bitstream with Processor Data

Double click on Assign Package Pins

Floorplan I/O - Presynthesis
(with ISE WebPack 10.1)

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;

entity top_level is
    port( l_r_haz :in std_logic_vector(2 downto 0);
          clk      : in std_logic;
          rst      : in std_logic;
          lights   : out std_logic_vector(5 downto 0);
          clkdiv   : out std_logic
    );
end top_level;

architecture struct of top_level is
    component counter
        port (clk, rst : in std_logic;
              countOut : out std_logic
        );
    end component;

    clk      : in std_logic;
    rst      : in std_logic;
    lights   : out std_logic_vector(5 downto 0)
    );
end component;
--signal declaration

signal countOut : std_logic;
```

Started : "Launching ISE Text Editor to edit counter.vhd".

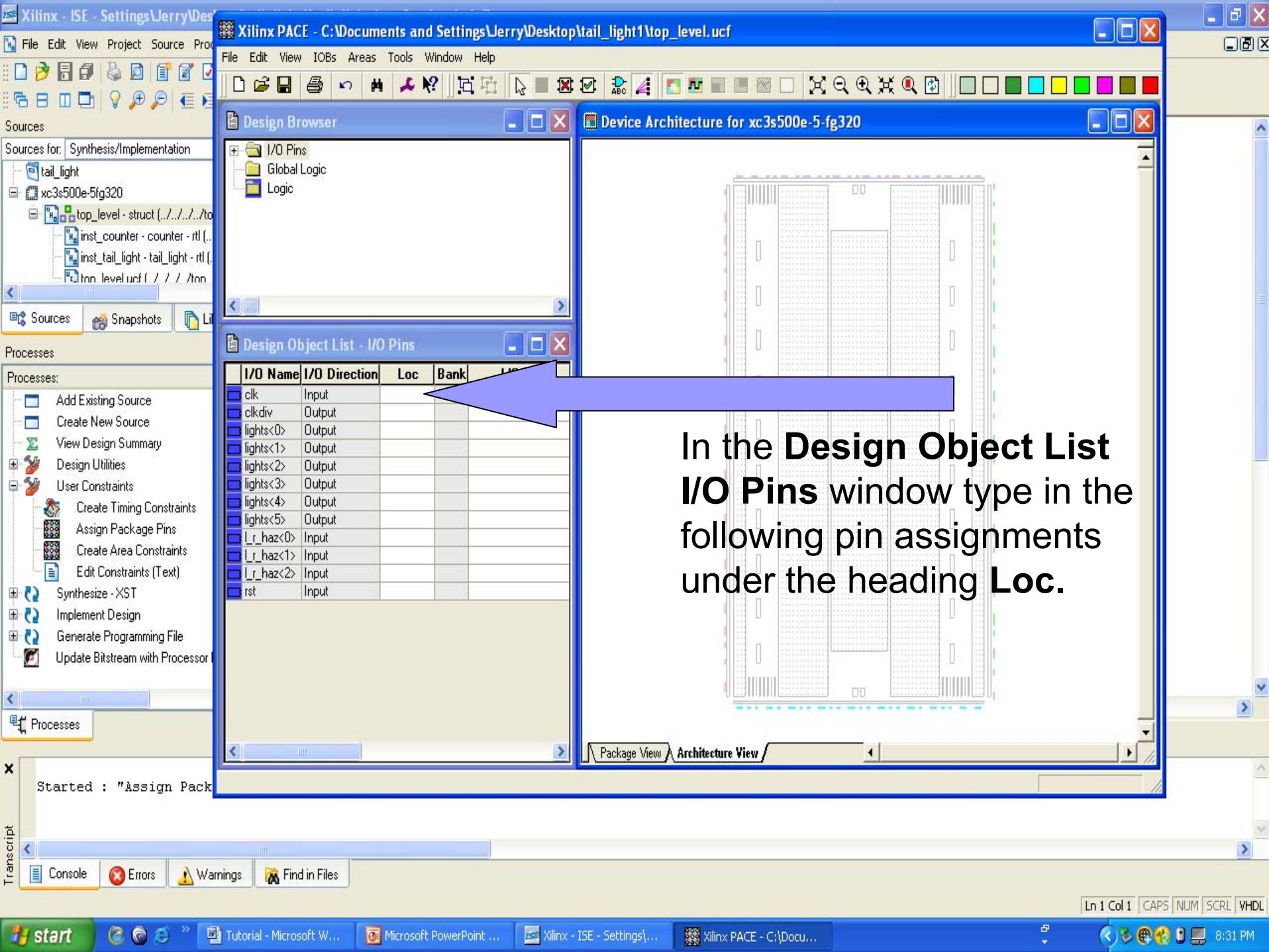
Started : "Launching ISE Text Editor to edit tail_light.vhd".

Console Errors Warnings Find in Files

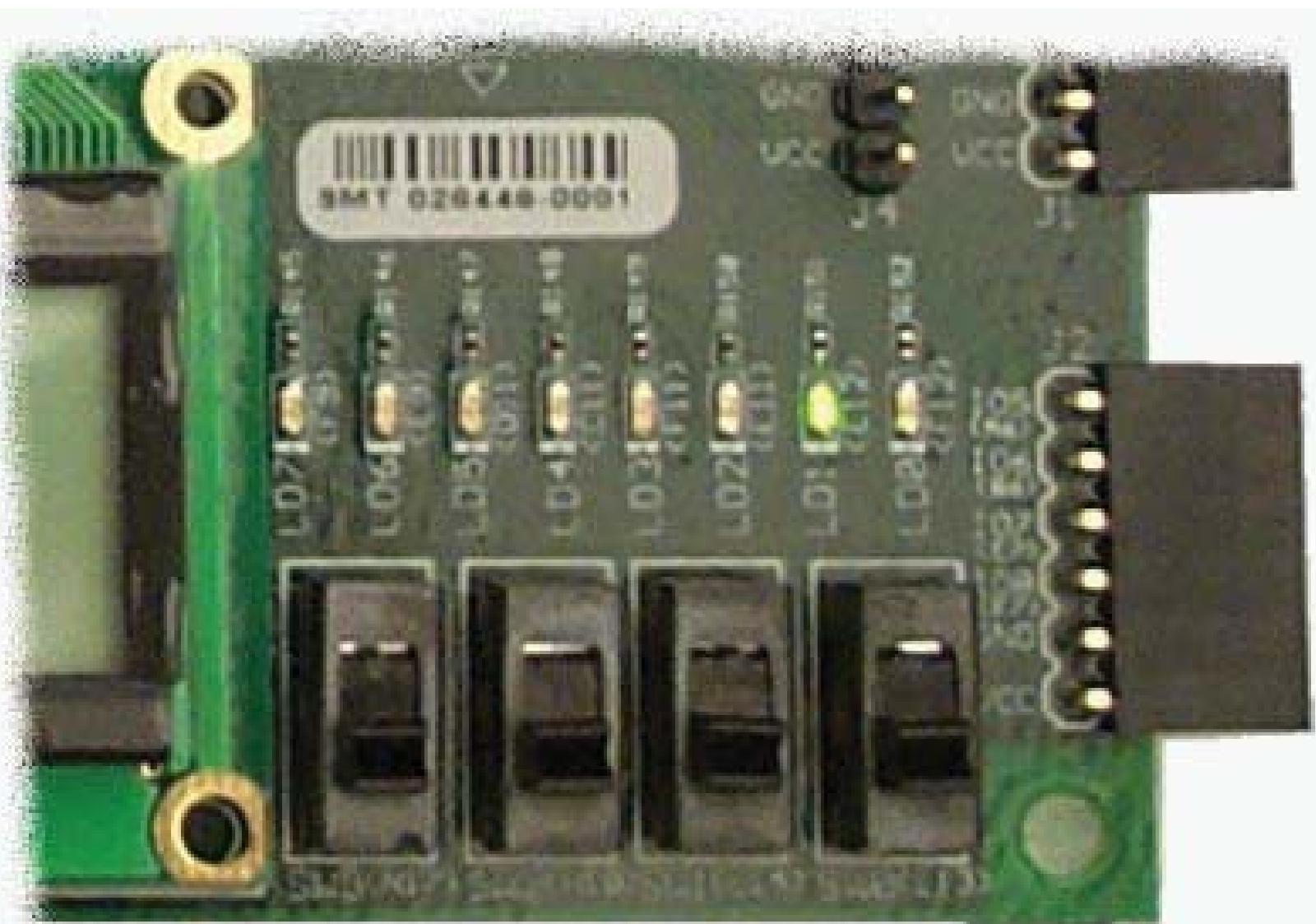
Ln 1 Col 1 CAPS NUM SCRL VHDL

start Tutorial - Microsoft W... Microsoft PowerPoint ... Xilinx - ISE - Settings\...

8:30 PM

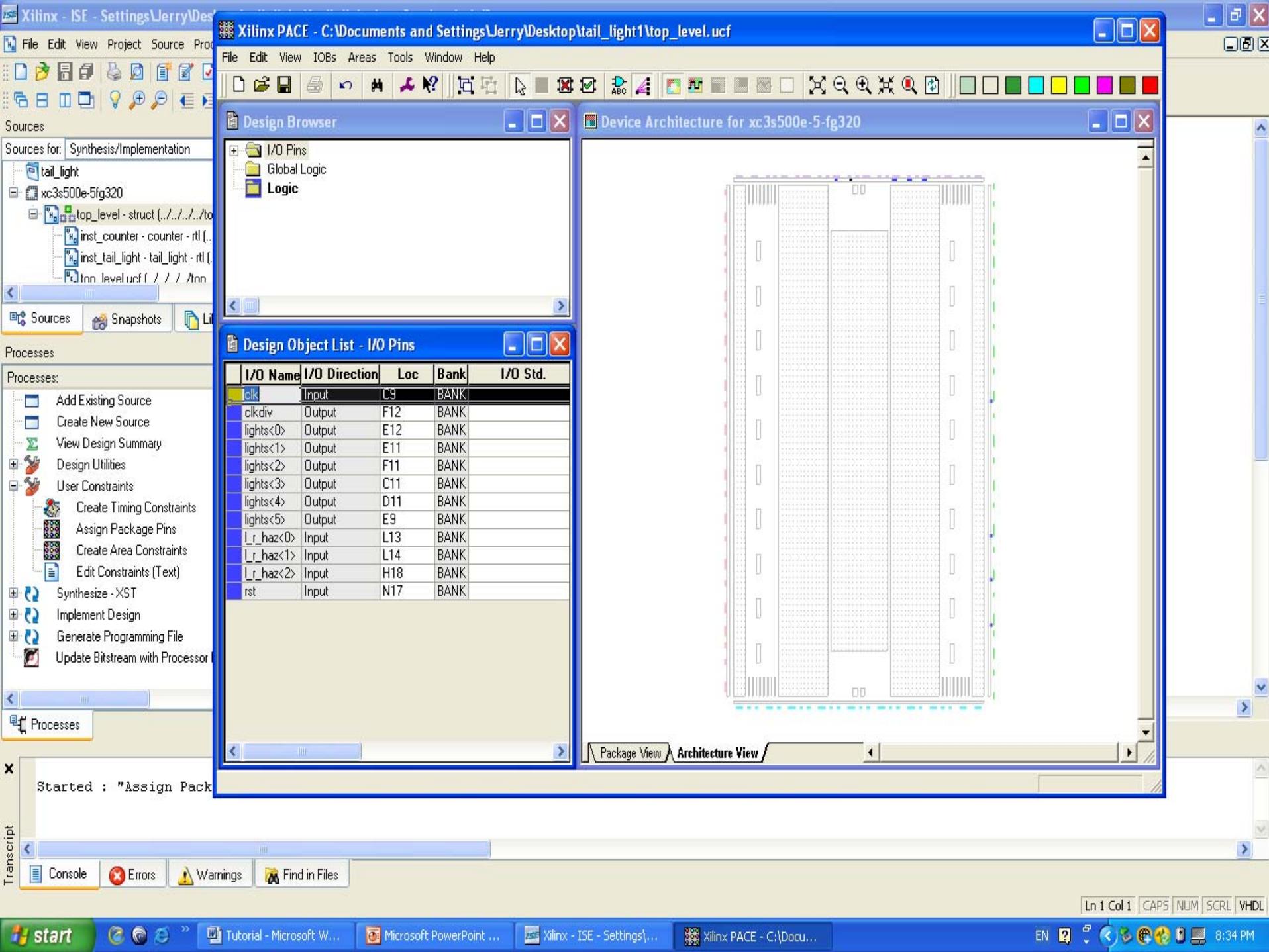


Four Slide Switches

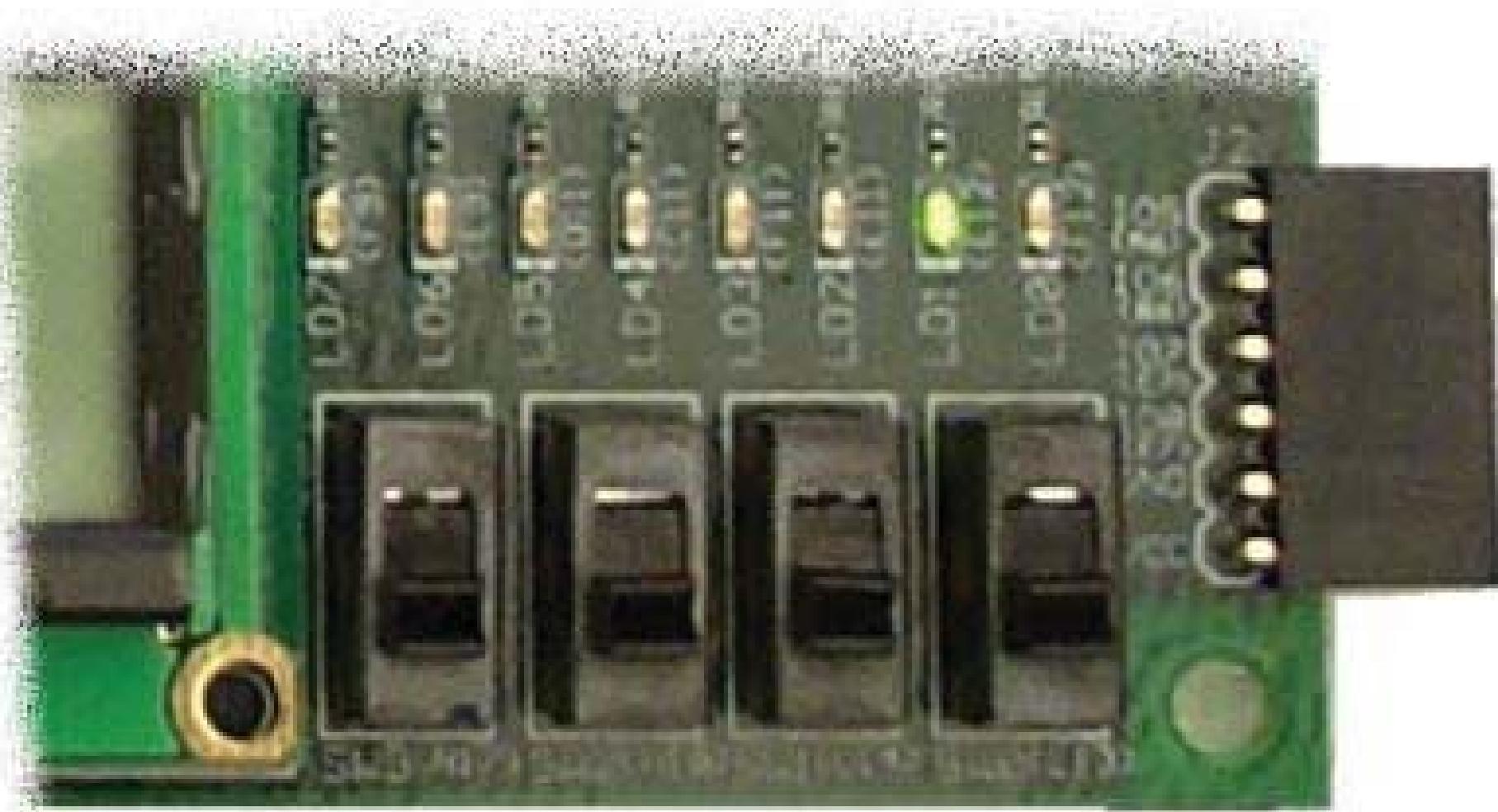


Pin assignments

- NET "SW<0>" LOC = "L13" | IOSTANDARD = LVTTL | PULLUP ;
- NET "SW<1>" LOC = "L14" | IOSTANDARD = LVTTL | PULLUP ;
- NET "SW<2>" LOC = "H18" | IOSTANDARD = LVTTL | PULLUP ;
- NET "SW<3>" LOC = "N17" | IOSTANDARD = LVTTL | PULLUP ;

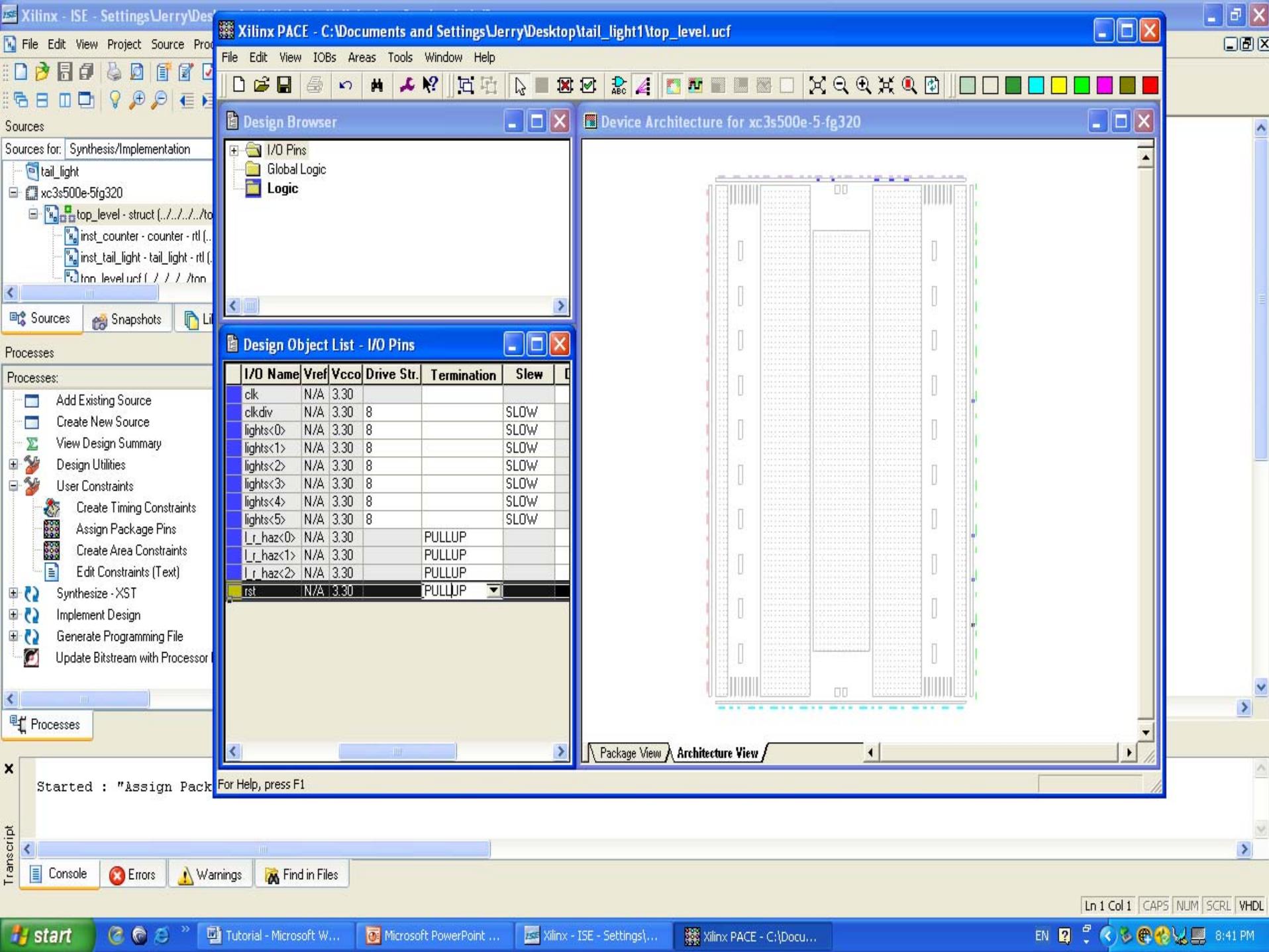


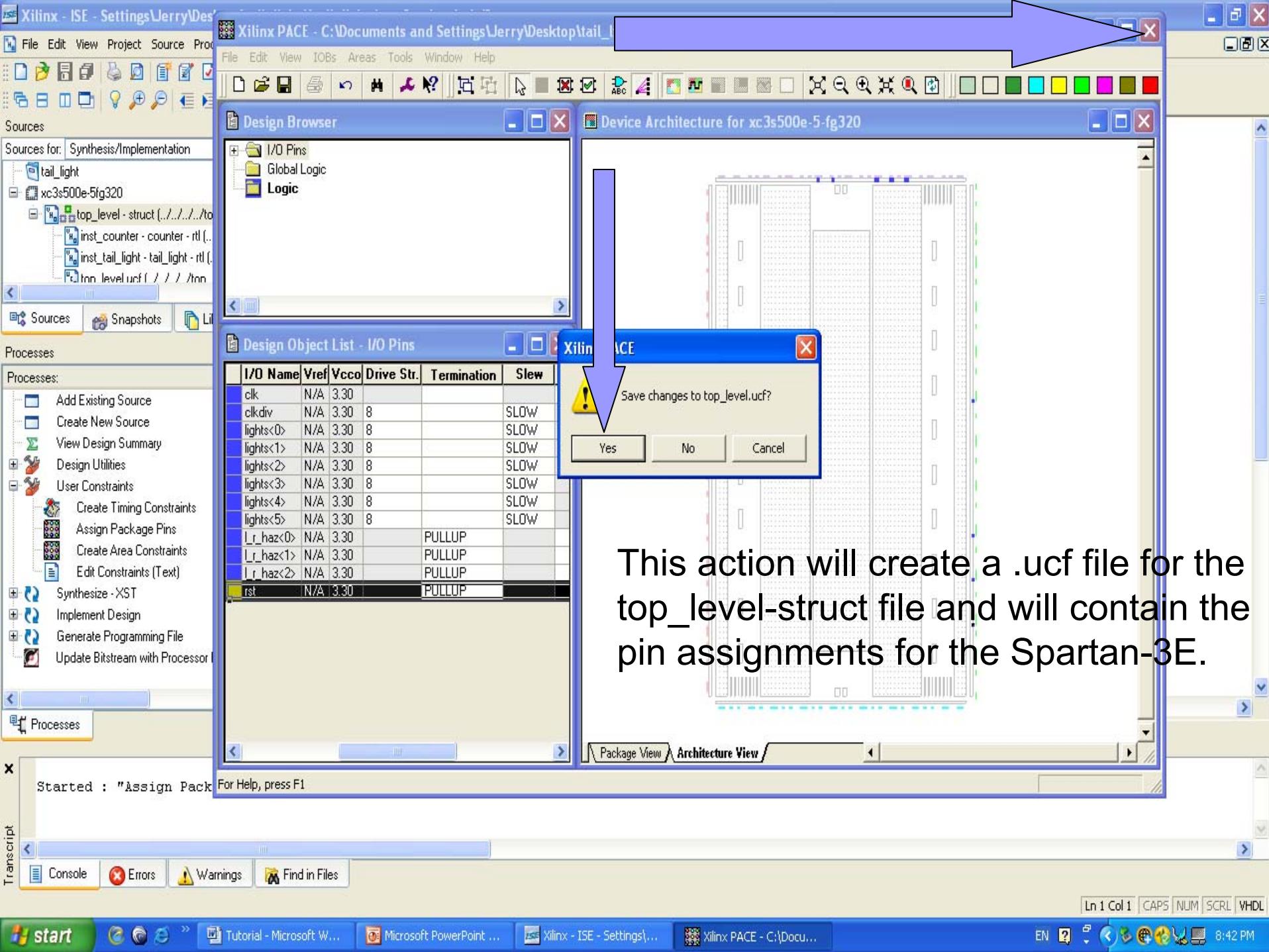
Eight Discrete LEDs



UCF Constraints for Eight Discrete LEDs

- NET "LED<7>" LOC = "F9" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
- NET "LED<6>" LOC = "E9" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
- NET "LED<5>" LOC = "D11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
- NET "LED<4>" LOC = "C11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
- NET "LED<3>" LOC = "F11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
- NET "LED<2>" LOC = "E11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
- NET "LED<1>" LOC = "E12" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
- NET "LED<0>" LOC = "F12" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;





This action will create a .ucf file for the top_level-struct file and will contain the pin assignments for the Spartan-3E.

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources

Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - struct (../../../../toplevel.vhd)
- inst_counter - counter - rtl (../../../../rtl/counter.vhd)
- inst_tail_light - tail_light - rtl (../../../../rtl/tail_light.vhd)
- inn_level_if (// / / inn_level_if.vhd)

Processes

Processes:

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Synthesize-XST**
- Implement Design
- Generate Programming File
- Update Bitstream with Processor Data

toplevel

counter

tail_light

Mapping all equations...
Building and optimizing final netlist ...
Found area constraint ratio of 100 (+ 5) on block top_level, actual ratio is 0.

Console Errors Warnings Find in Files

1000 ns

1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.std_logic_arith.all;
4
5 clk : in std_logic;
6 rst : in std_logic;
7 lights : out std_logic_vector(5 downto 0);
8 clkdiv : out std_logic
9);
10 end top_level;
11
12 architecture struct of top_level is
13 component counter
14 port (clk, rst : in std_logic;
15 countOut : out std_logic);
16 end component;
17 --signal declaration
18
19 component tail_light
20 port (l_r_haz : in std_logic_vector(2 downto 0);
21 clk : in std_logic;
22 rst : in std_logic;
23 lights : out std_logic_vector(5 downto 0);
24);
25 end component;
26 --signal declaration
27
28 signal countOut : std_logic;
29
30

Make sure that **top_level-struct** is selected in the **Sources** window.

In the **Processes** window, expand the **Synthesize-XST** process and then double click it. This action will check the syntax of the source code for **top_level-struct** and convert the source code into a netlist of gates. A synthesis report will also be produced. When the synthesis is finished, green check marks should be displayed just like in the next slide indicating that **top_level-struct** has compiled successfully.

Ln 1 Col 1 CAPS NUM SCR VHD

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - struct (../../toplevel)
- inst_counter - counter - rtl (../../inst_counter)
- inst_tail_light - tail_light - rtl (../../inst_tail_light)
- inn_level_if (// inn level if)

Processes

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Synthesize - XST**
- Implement Design
- Generate Programming File
- Update Bitstream with Processor Data

1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.std_logic_arith.all;
4
5 entity top_level is
6 port(l_r_haz : in std_logic_vector(2 downto 0);
7 clk : in std_logic;
8 rst : in std_logic;
9 lights : out std_logic_vector(5 downto 0);
10 clkdiv : out std_logic
11);
12 end top_level;
13
14 architecture struct of top_level is
15 component counter
16 port(clk, rst : in std_logic;
17 countOut : out std_logic
18);
19 end component;
20
21 port(l_r_haz : in std_logic_vector(2 downto 0);
22 clk : in std_logic;
23 rst : in std_logic;
24 lights : out std_logic_vector(5 downto 0)
25);
26 end component;
27 --signal declaration
28
29 signal countOut : std_logic;
30

Processes

toplevel counter tail_light

WARNING:ProjectMgmt - "C:/Documents and Settings/Jerry/Desktop/tail_light1/top_level.ngr" line 0 duplicate design unit: 'Module|top_level'

Process "Synthesize" completed successfully

Transcript

Console Errors Warnings Find in Files

Ready

Ln 1 Col 1 CAPS NUM SCRL VHDL

start Tutorial - Microsoft W... Microsoft PowerPoint ... Xilinx - ISE - Settings...

8:43 PM

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources Sources for: Synthesis/Implementation

tail_light xc3s500e-5fg320 top_level - struct (././././toplevel) inst_counter - counter - rtl (./././inst_tail_light - tail_light - rtl (./././inst_level.vdf (./././inst_level))

Processes

Add Existing Source Create New Source View Design Summary Design Utilities User Constraints Synthesize - XST Implement Design Generate Programming File Update Bitstream with Processor Data

1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.std_logic_arith.all;
4
5 entity top_level is
6 port(l_r_haz :in std_logic_vector(2 downto 0);
7 clk : in std_logic;
8 rst : in std_logic;
9 lights : out std_logic_vector(5 downto 0);
10 clkdiv : out std_logic
11);
12 end top_level;
13
14 architecture struct of top_level is
15 component counter
16 port (clk, rst : in std_logic;
17 countOut : out std_logic
18);
19 end component;
20
21 clk : in std_logic;
22 rst : in std_logic;
23 lights : out std_logic_vector(5 downto 0)
24);
25 end component;
26 --signal declaration
27
28 signal countOut : std_logic;

toplevel counter tail_light

Reading NGO file 'C:/Documents and Settings/Jerry/Desktop/tail_light1/top_level.ngc' ...

Console Errors Warnings Find in Files

Ready

Ln 1 Col 1 CAPS NUM SCRL VHDL

start Tutorial - Microsoft W... Microsoft PowerPoint ... Xilinx - ISE - Settings\...

8:44 PM

Expand the Implement Design process and double click on it. This is where the netlist is translated, mapped, placed and routed for the logic circuits of the Spartan-3E FPGA.

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources Sources for: Synthesis/Implementation

tail_light
xc3s500e-5fg320
top_level - struct (../../toplevel.vhd)

inst_counter - counter - rtl (../../rtl/inst_counter.vhd)
inst_tail_light - tail_light - rtl (../../rtl/inst_tail_light.vhd)
inn_level_if (// inn level_if.vhd)

Sources Snapshots Lib

Processes

Add Existing Source
Create New Source
View Design Summary
Design Utilities
User Constraints
Synthesize - XST
Implement Design

Generate Programming File
Update Bitstream with Processor Data

1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.std_logic_arith.all;
4
5 entity top_level is
6 port(l_r_haz :in std_logic_vector(2 downto 0);
7 clk : in std_logic;
8 rst : in std_logic;
9 lights : out std_logic_vector(5 downto 0);
10 clkdiv : out std_logic
11);
12 end top_level;
13
14 architecture struct of top_level is
15 component counter
16 port (clk, rst : in std_logic;
17 countOut : out std_logic
18);
19 end component;
20
21
22 clk : in std_logic;
23 rst : in std_logic;
24 lights : out std_logic_vector(5 downto 0)
25);
26 end component;
27 --signal declaration
28
29 signal countOut : std_logic;
30

toplevel counter tail_light

Total time: 3 secs
Process "Generate Post-Place & Route Static Timing" completed successfully

Console Errors Warnings Find in Files

Ready

Ln 1 Col 1 CAPS NUM SCRL VHDL

start Tutorial - Microsoft W... Microsoft PowerPoint ... Xilinx - ISE - Settings\... 8:44 PM

After this process has been run, green check marks should be displayed

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources

Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - struct (../../../../toplevel.vhd)
- inst_counter - counter - rtl (../../rtl/inst_counter.vhd)
- inst_tail_light - tail_light - rtl (../../rtl/inst_tail_light.vhd)
- inn_level_if (// / inn_level_if.vhd)

Processes

Processes:

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Synthesize - XST
- Implement Design**
- Generate Programming File
 - Programming File Generation Report
 - Generate PROM, ACE, or JTAG File
 - Configure Device (IMPACT)
 - Update Bitstream with Processor Data

Expand the **Generate Programming File** process and double click on it.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;

entity top_level is
    port( l_r_haz : in std_logic_vector(2 downto 0);
          clk      : in std_logic;
          rst      : in std_logic;
          lights   : out std_logic_vector(5 downto 0);
          clkdiv  : out std_logic
    );
end top_level;

architecture struct of top_level is
begin
    component counter
        port (clk, rst : in std_logic;
              countOut : out std_logic
        );
    end component;

    component tail_light
        port( l_r_haz : in std_logic_vector(2 downto 0);
              lights   : out std_logic_vector(5 downto 0)
        );
    end component;
    --signal declaration

    signal countOut : std_logic;

```

Total time: 3 secs

Process "Generate Post-Place & Route Static Timing" completed successfully

Console Errors Warnings Find in Files

Ln 1 Col 1 CAPS NUM SCRL VHDL

Ready

start Tutorial - Microsoft W... Microsoft PowerPoint ... Xilinx - ISE - Settings\...

8:45 PM

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources

Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - struct (../../../../toplevel.vhd)
- inst_counter - counter - rtl (../../../../counter.vhd)
- inst_tail_light - tail_light - rtl (../../../../tail_light.vhd)
- inn_level_if (// / inn level_if.vhd)

Processes

Processes:

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Synthesize - XST
- Implement Design
- Generate Programming File
- Programming File Generation Report
- Generate PROM, ACE, or JTAG File
- Configure Device (IMPACT)
- Update Bitstream with Processor Data

Script

Started : "Generate Programming File".

Process "Generate Programming File" completed successfully

Console Errors Warnings Find in Files

1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.std_logic_arith.all;
4
5 entity top_level is
6 port(l_r_haz : in std_logic_vector(2 downto 0);
7 clk : in std_logic;
8 rst : in std_logic;
9 lights : out std_logic_vector(5 downto 0);
10 clkdiv : out std_logic
11);
12 end top_level;
13
14 architecture struct of top_level is
15 component counter
16 port (clk, rst : in std_logic;
17 countOut : out std_logic
18);
19 end component;
20
21 component tail_light
22 port(l_r_haz : in std_logic_vector(2 downto 0);
23 lights : out std_logic_vector(5 downto 0)
24);
25 end component;
26 --signal declaration
27
28 signal countOut : std_logic;

A large blue arrow points from the "Generate Programming File" option in the Processes list to the corresponding code in the main editor window.

This process creates a bit file that is used to program the Spartan-3E chip.
Again, after this process is finished, green check marks should be shown.

Ln 1 Col 1 CAPS NUM SCRL VHDL

FPGA

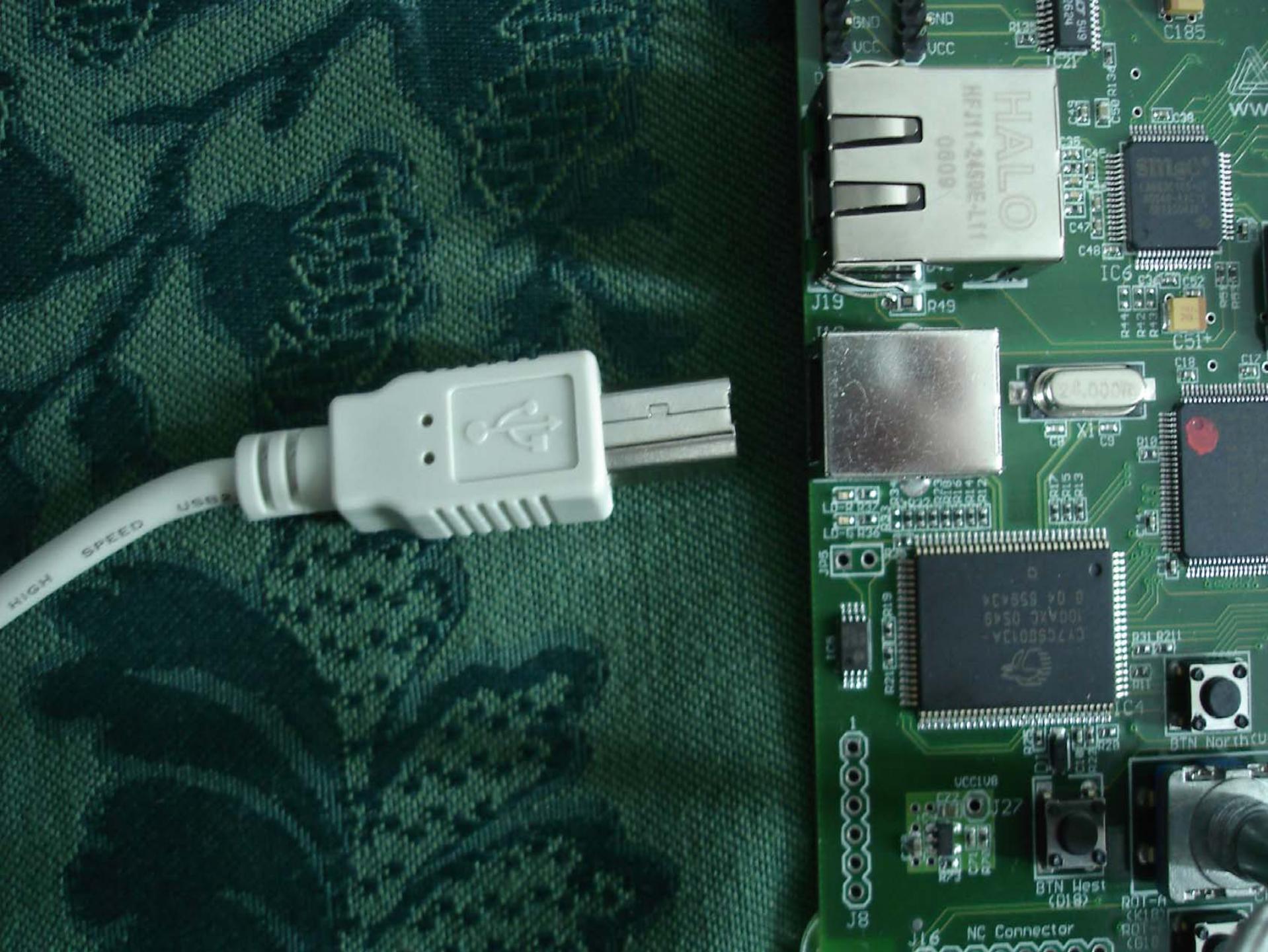
- Now you are ready to program the Spartan-3E chip. Plug-in the USB cable to the computer. Connect the FPGA Spartan-3E board to power **FIRST**, then connect the USB cable to the board. This order must be followed for the Spartan-3E chip to be programmed properly.

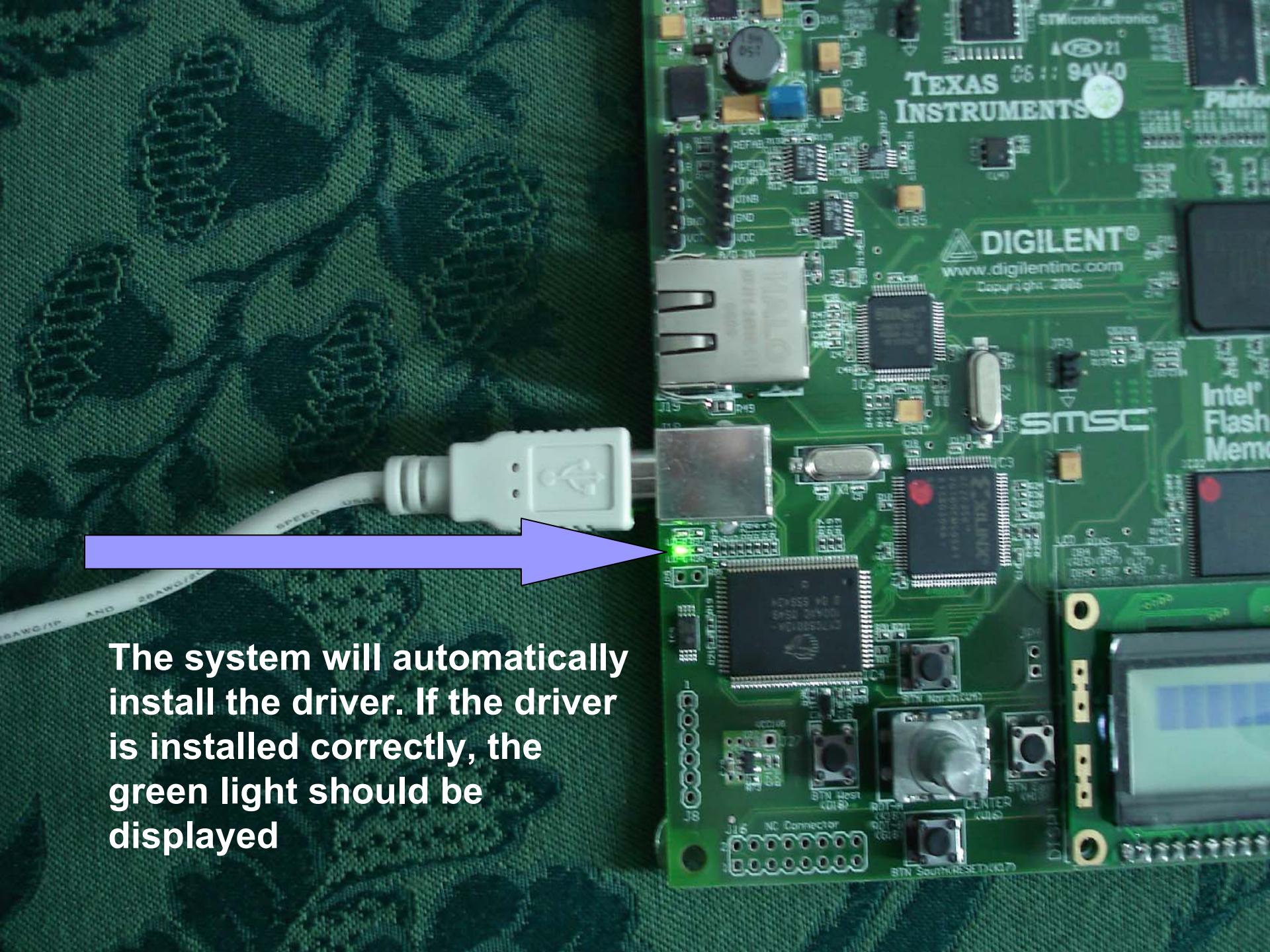




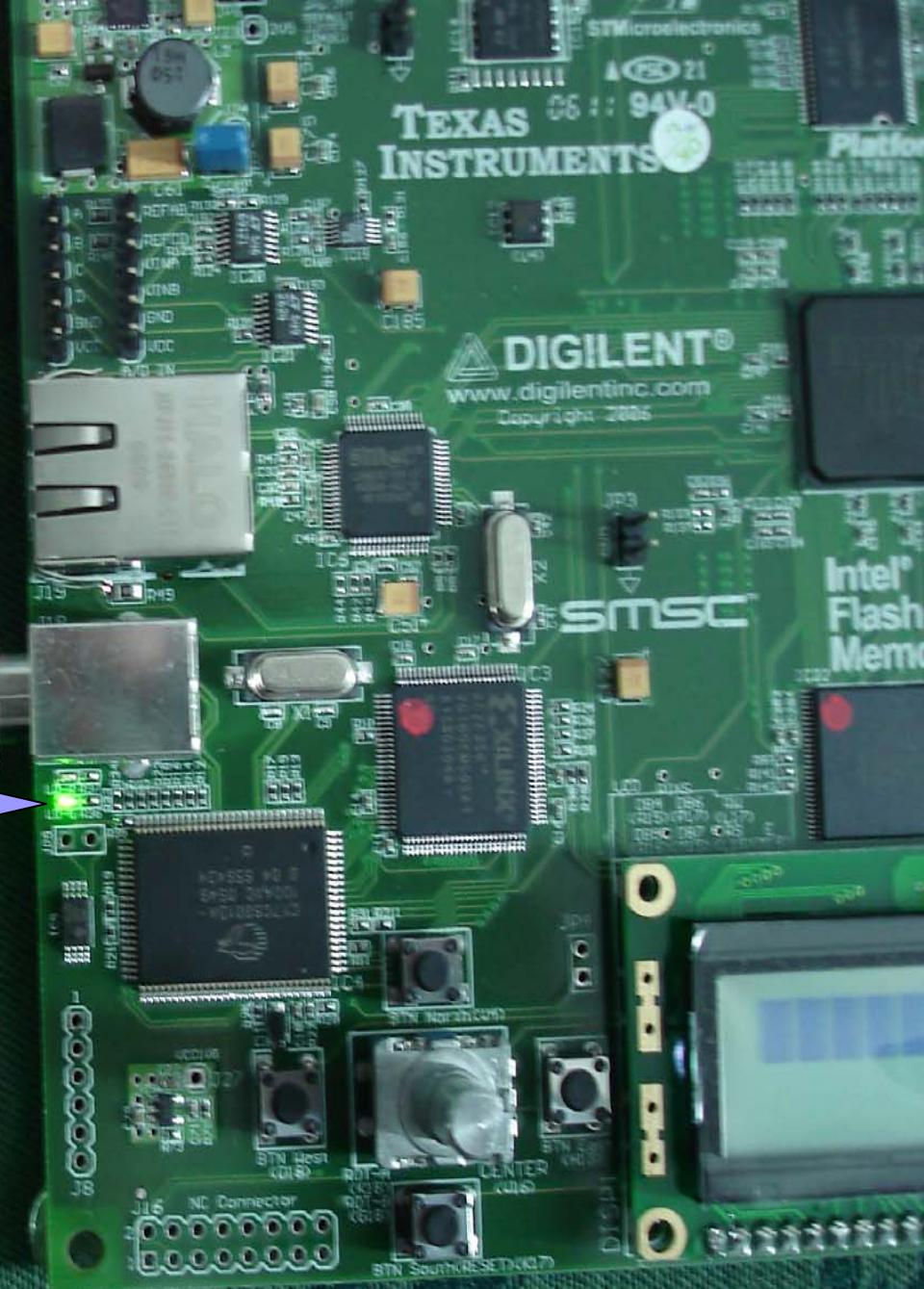
POWER is ON







The system will automatically install the driver. If the driver is installed correctly, the green light should be displayed



Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources Sources for: Synthesis/Implementation

tail_light
xc3s500e-5fg320
top_level - struct (././././toplevel)
inst_counter - counter - rtl (././.)
inst_tail_light - tail_light - rtl (././.)
inn_level_if (/ / / inn level)

Sources Snapshots Lib

Processes Processes:

- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Synthesize - XST
- Implement Design
- Generate Programming File
- Programming File Generation Report
- Generate PROM, ACE, or JTAG File
- Configure Device (iMPACT)
- Update Bitstream with Processor Data

Now double click on **Configure Device (iMPACT)** under the **Generate Programming File** process.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;

entity top_level is
    port( l_r_haz :in std_logic_vector(2 downto 0);
          clk      :in std_logic;
          rst      :in std_logic;
          lights   :out std_logic_vector(5 downto 0);
          clkdiv  :out std_logic
    );
end top_level;

architecture struct of top_level is
begin
    component counter
        port (clk, rst : in std_logic;
              countOut : out std_logic
        );
    end component;

    component tail_light
        port ( l_r_haz :in std_logic_vector(2 downto 0);
              clk      :in std_logic;
              rst      :in std_logic;
              lights   :out std_logic_vector(5 downto 0)
        );
    end component;

    signal countOut : std_logic;
    --signal declaration
end;
```

toplevel counter tail_light

Started : "Generate Programming File".

Process "Generate Programming File" completed successfully

Console Errors Warnings Find in Files

Ready

Ln 1 Col 1 CAPS NUM SCRL VHDL

start Tutorial - Microsoft W... Microsoft PowerPoint ... Xilinx - ISE - Settings\...

8:46 PM

Xilinx - ISE - Settings\Jerry\Desktop\tail_light1\tail_light.ise - [toplevel.vhd]

File Edit View Project Source Process Window Help

Sources Sources for: Synthesis/Implementation
tail_light
xc3s500e-5fg320
Flows top_level - struct (.../.../.../tople...
inst_counter - counter - rtl (.../.../.../inst...
inst_tail_light - tail_light - rtl (.../.../.../inst...
top_level.vhf (.../.../.../top_level.vhf
Sources Snapshots Lib

Processes Processes:
Add Existing Source
Create New Source
View Design Summary
Design Utilities
User Constraints
Synthesize -XST
Implement Design
Generate Programming File
Programming File Generation Report
Generate PROM, ACE, or JTAG File
Configure Device (iMPACT)
Update Bitstream with Processor Data

iMPACT - C:/Documents and Settings/Jerry/Desktop/tail_light1/Settings/Jerry/Desktop/tail_light1/tail_light.ipf

File Edit View Operations Options Output Debug Window Help

Flows Boundary Scan SlaveSerial SelectMAP Desktop Configuration SystemACE

iMPACT Modes
iMPACT Processes
iMPACT Process Operations
Welcome to iMPACT

Please select an action from the list below

Configure devices using Boundary-Scan (JTAG)

Automatically connect to a cable and identify Boundary-Scan chain

Prepare a PROM File

Prepare a System ACE File

Prepare a Boundary-Scan File

Configure devices

SVF
using Slave Serial mode

Finish Cancel

You will be asked how you want the device configured. Select **Boundary Scan Mode** and click Finish.

Process "Configure Device (iMPACT)" completed successfully

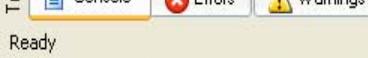
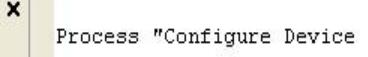
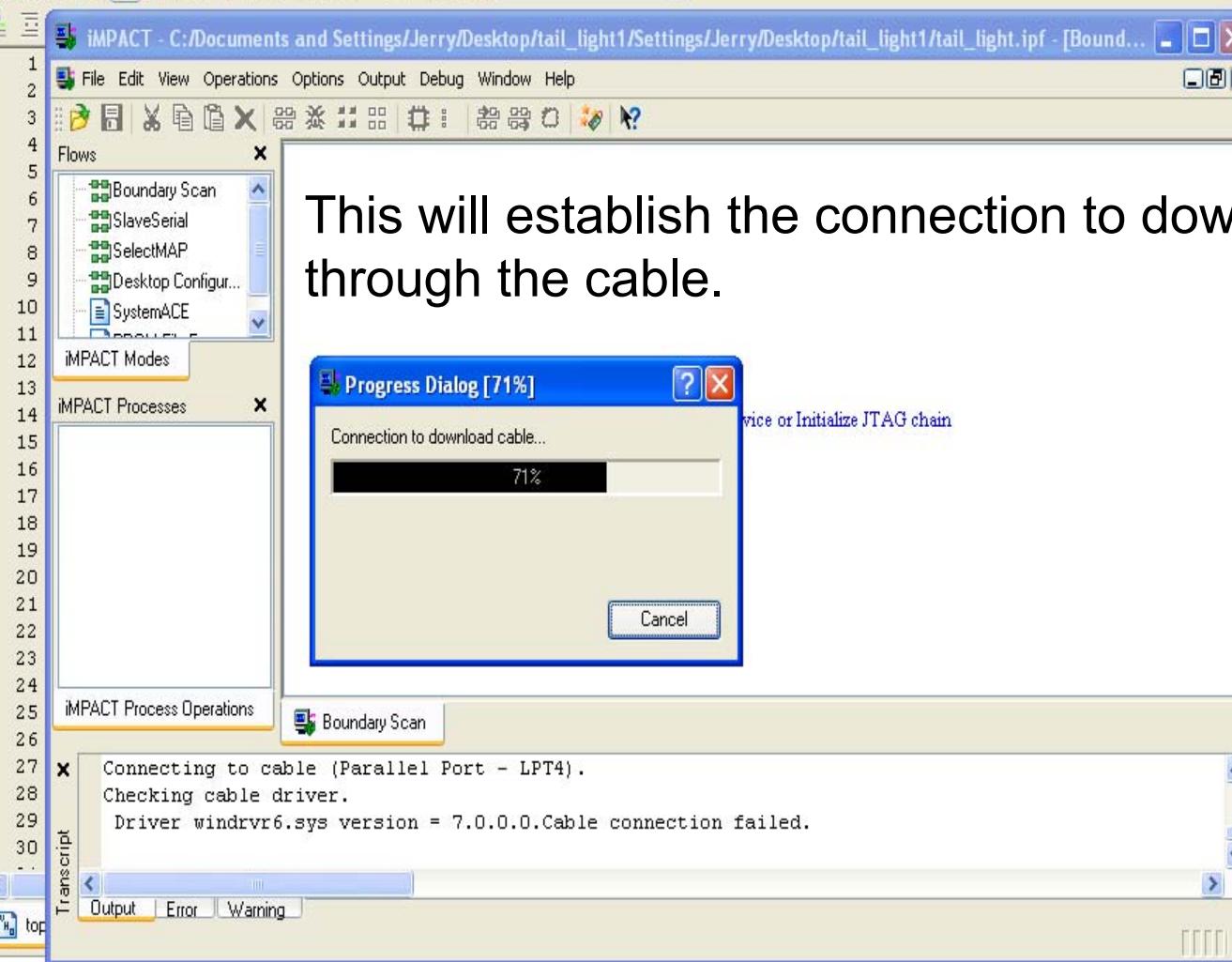
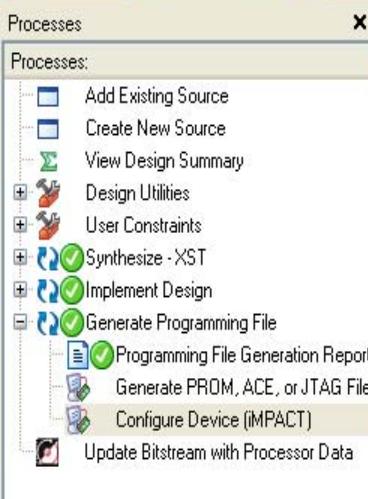
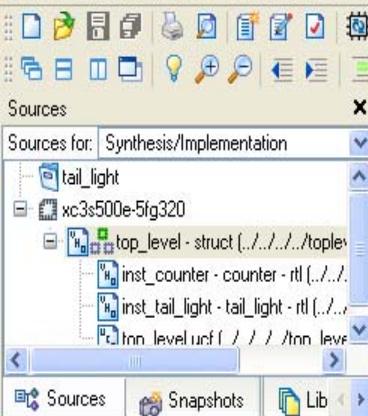
Console Errors Warnings Find in Files

Ready

Ln 1 Col 1 CAPS NUM SCRL VHDL

start Tutorial - Microsoft W... Microsoft PowerPoint ... Xilinx - ISE - Settings\... iMPACT - C:/Docume... EN 8:47 PM

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30





Sources

Sources for: Synthesis/Implementation

- tail_light
- xc3s500e-5fg320
- top_level - structure
 - inst_counter - counter
 - inst_tail_light - tail light
 - tnn_level udf f...

Sources

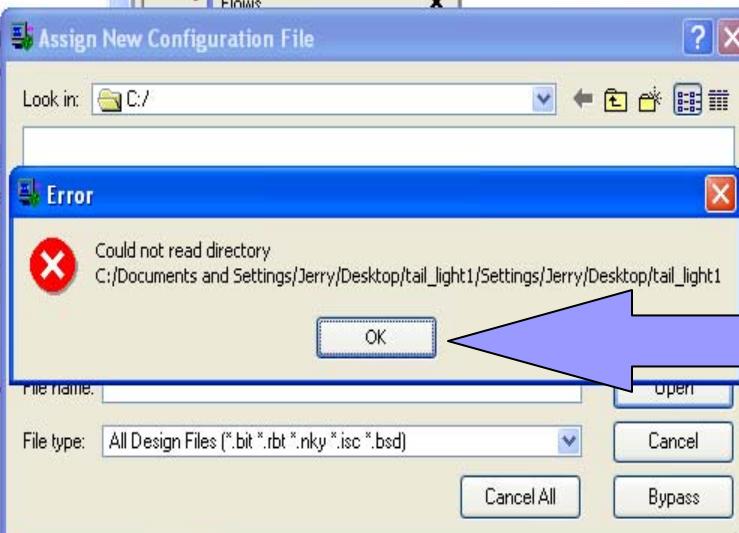
Processes

Processes:

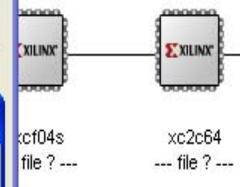
- Add Existing Source
- Create New Source
- View Design Summary
- Design Utilities
- User Constraints
- Synthesize - XST
- Implement Design
- Generate Programming File
 - Programming File Generation Report
 - Generate PROM, ACE, or JTAG File
 - Configure Device (iMPACT)
 - Update Bitstream with Processor Data

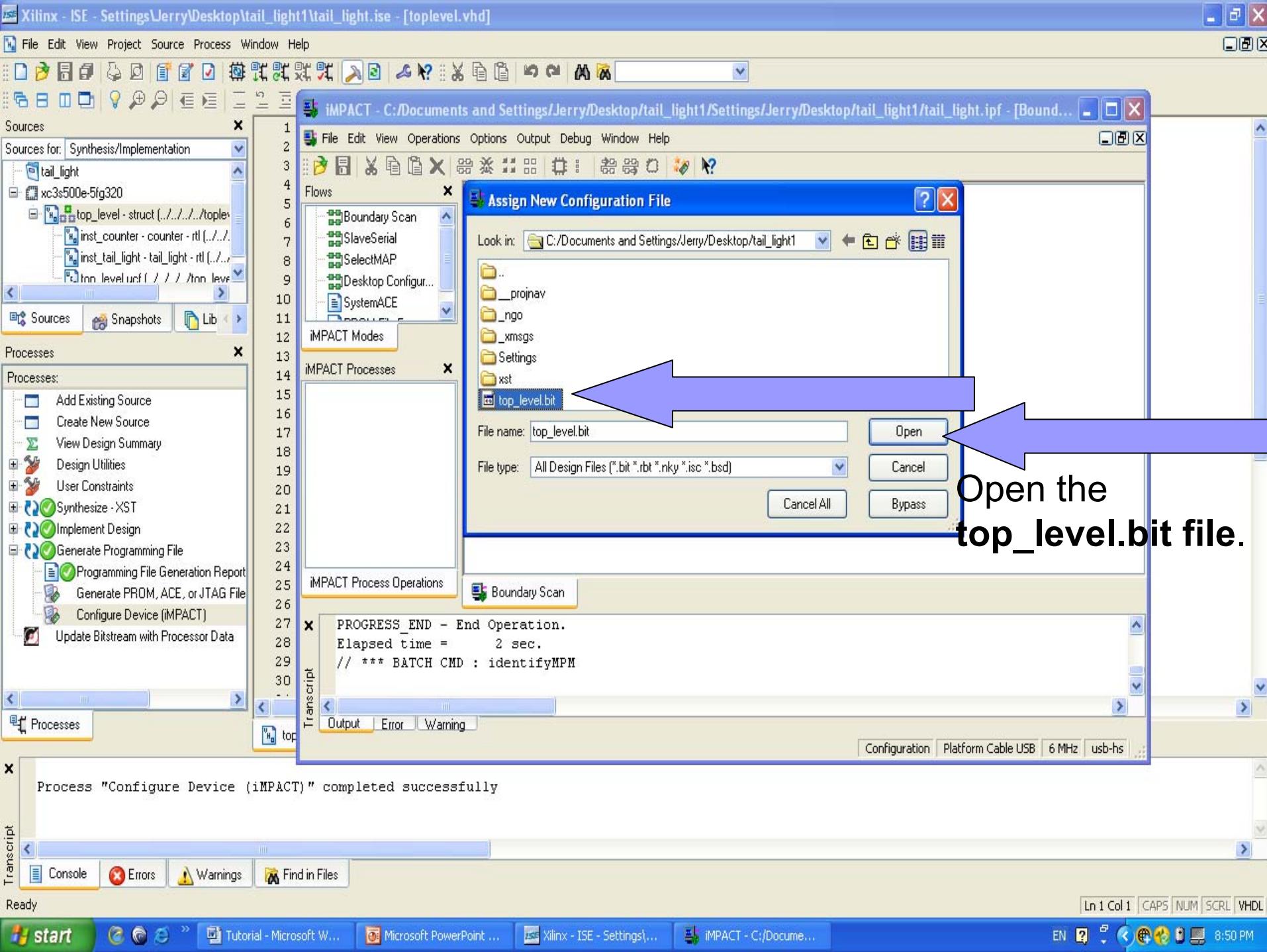
Processes

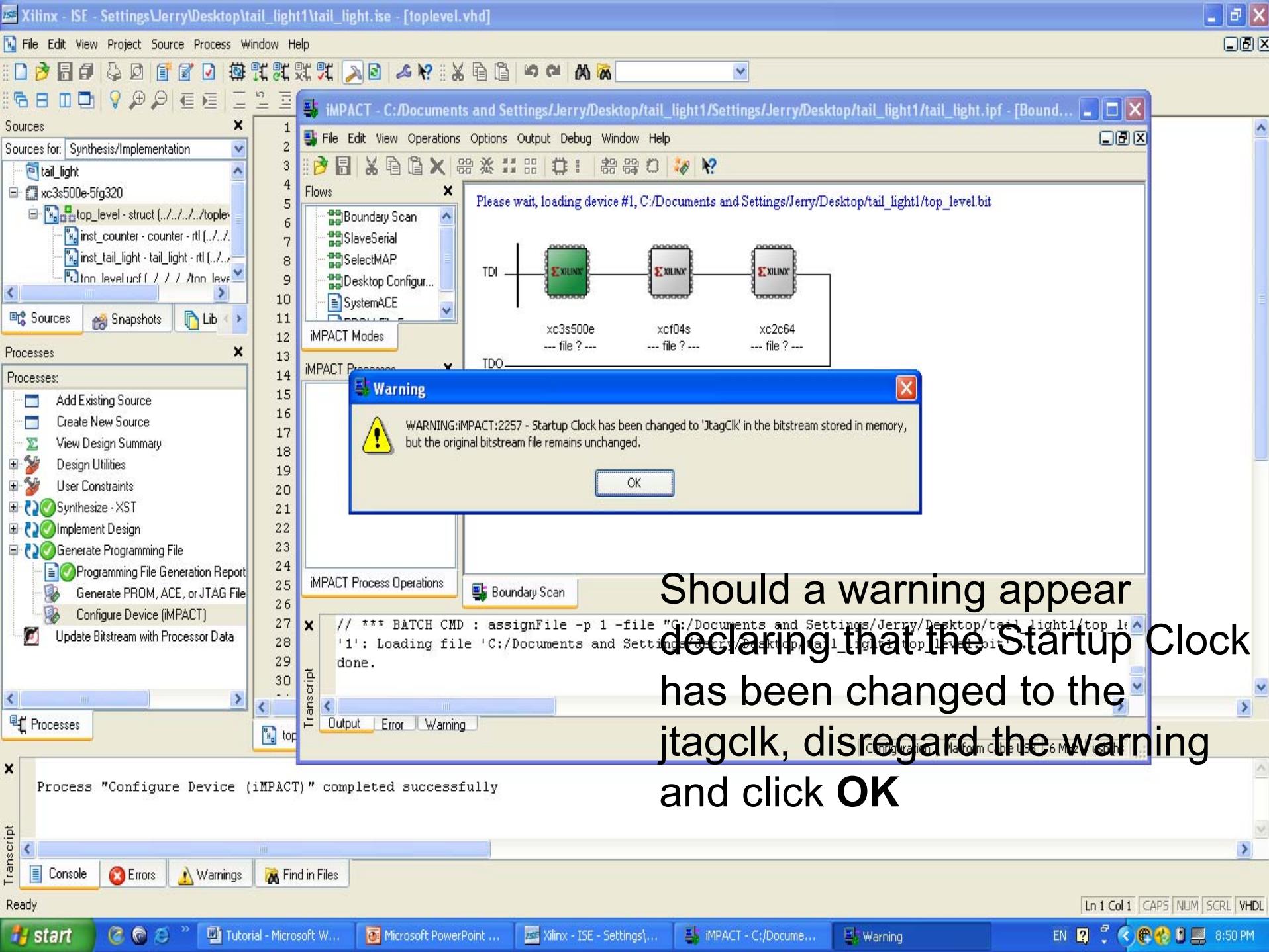
Process "Configure Device (iMPACT)" completed successfully

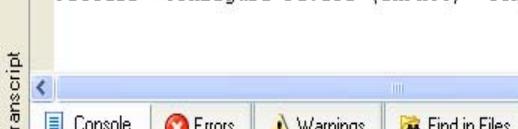
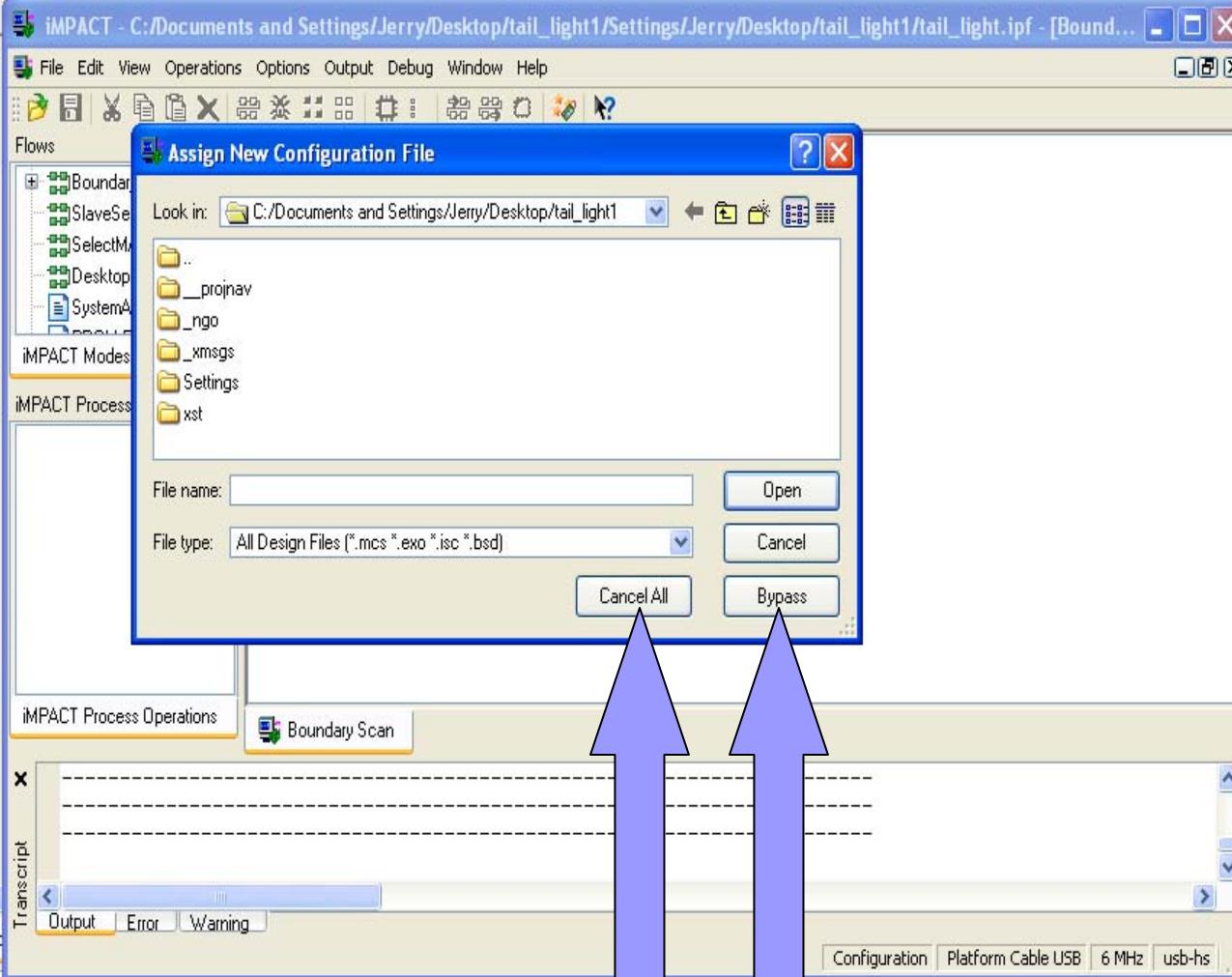
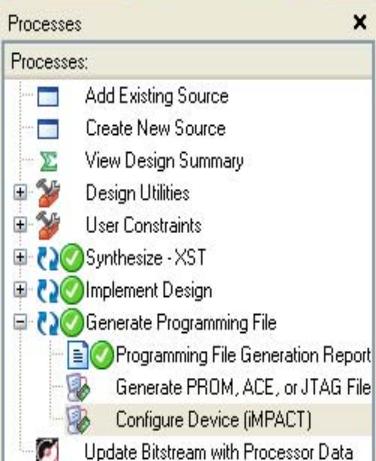
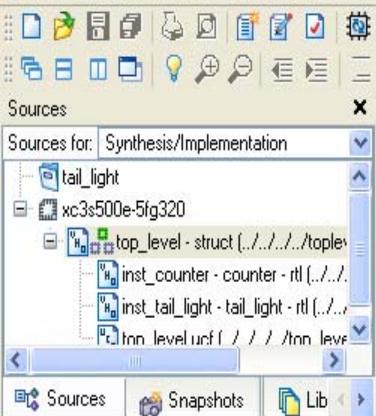


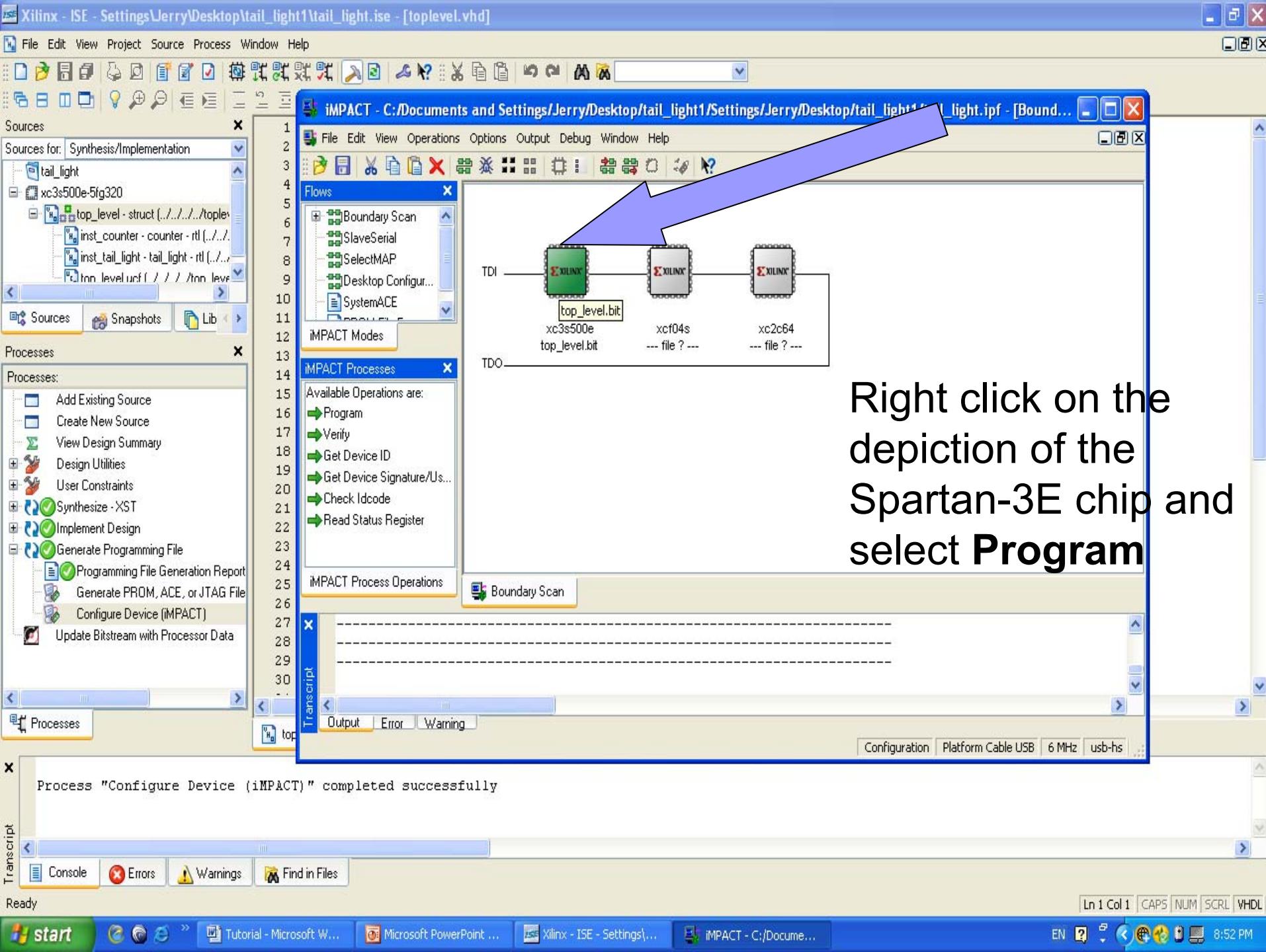
Identify Succeeded



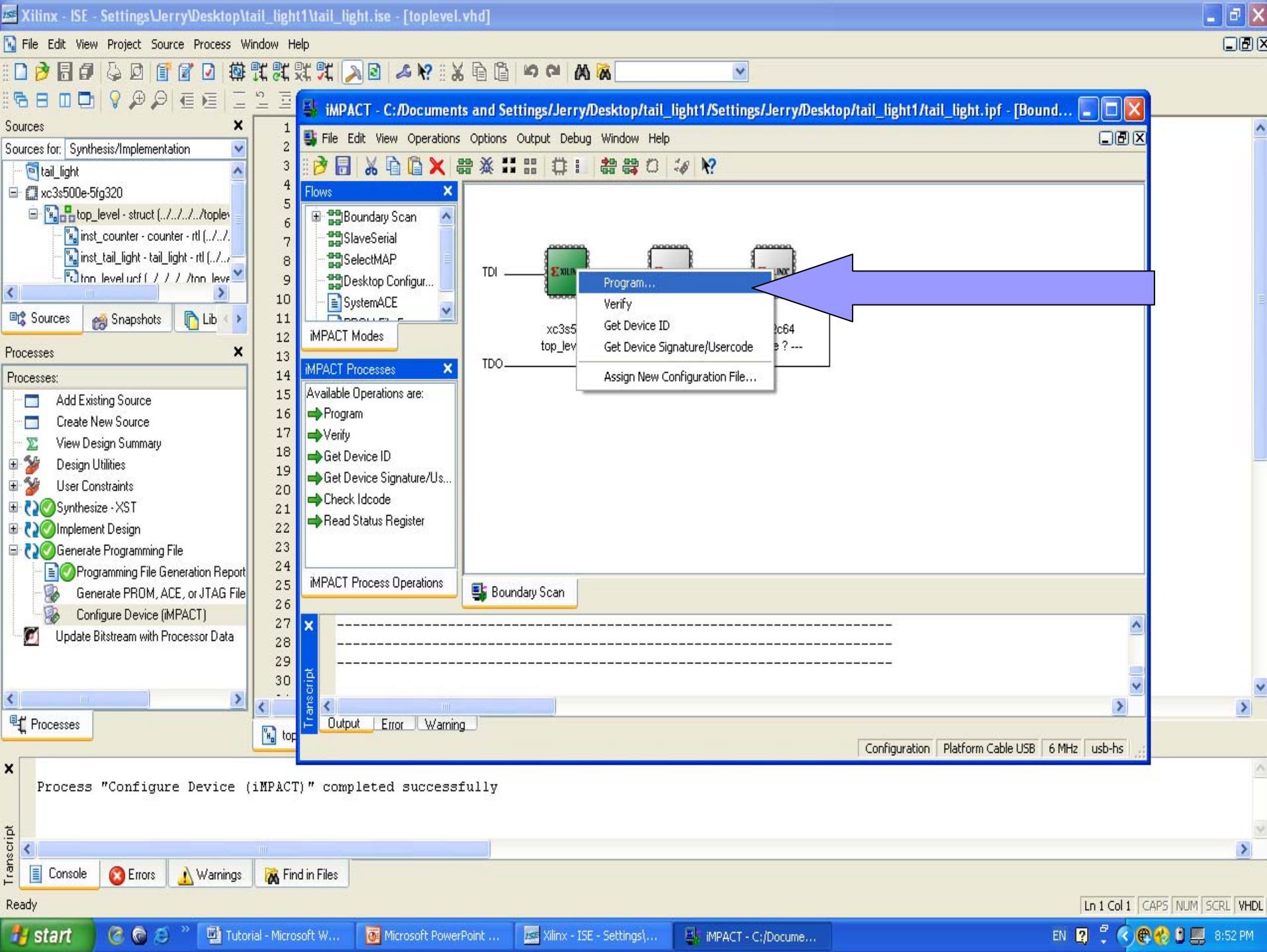


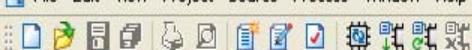






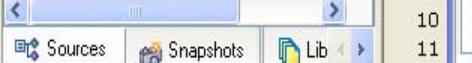
Right click on the
depiction of the
Spartan-3E chip and
select **Program**





Sources

- Sources for: Synthesis/Implementation
- tail_light
 - xc3s500e-5fg320
 - top_level - struct (../../../../top_level)
 - inst_counter - counter - rtl (../../../../inst_counter)
 - inst_tail_light - tail_light - rtl (../../../../inst_tail_light)
 - tnn_level udf (../../../../tnn_level)

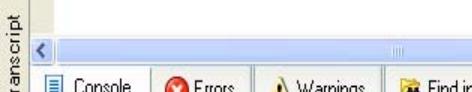


Processes

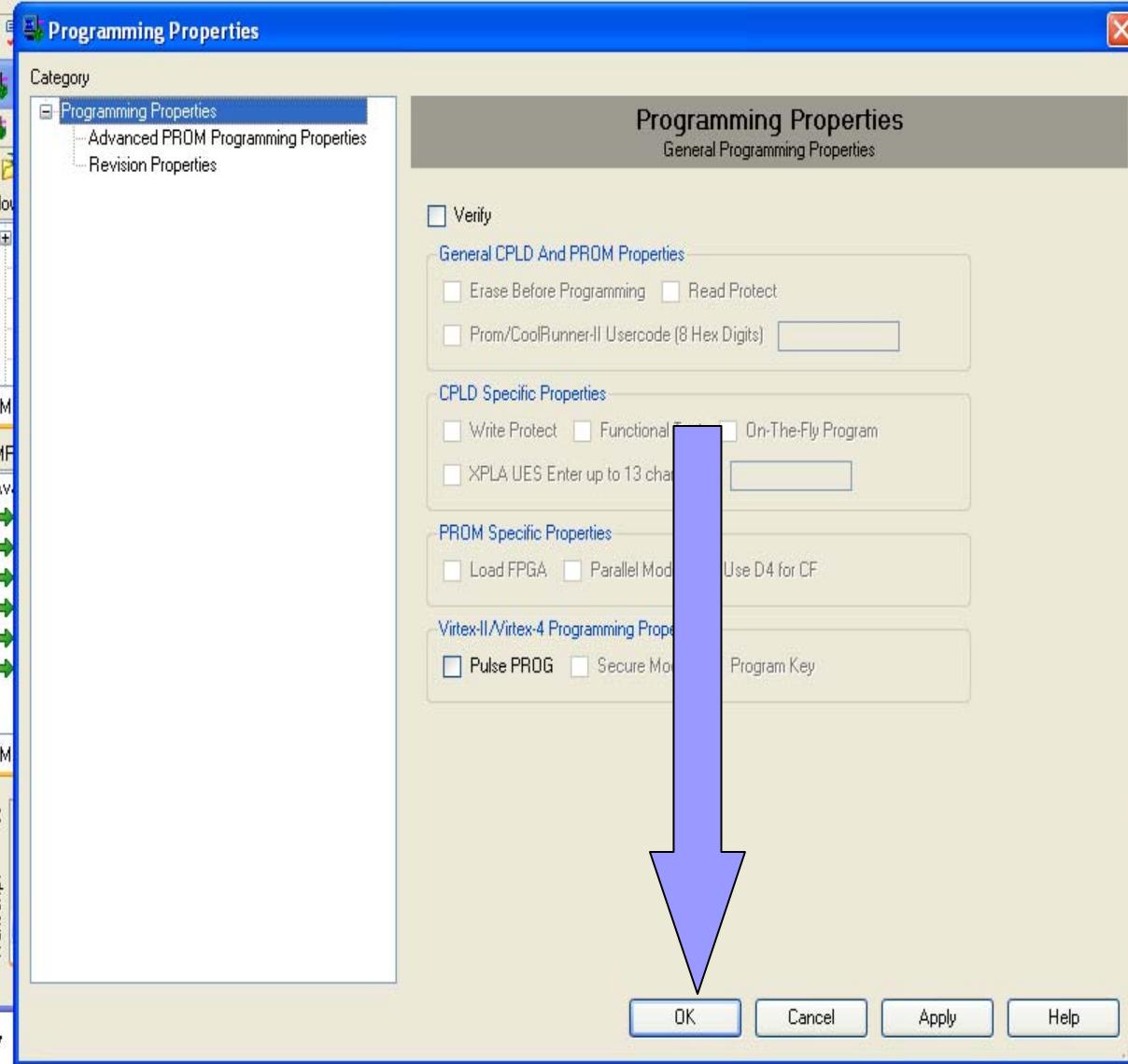
- Processes:
- Add Existing Source
 - Create New Source
 - View Design Summary
 - Design Utilities
 - User Constraints
 - Synthesizes - XST
 - Implement Design
 - Generate Programming File
 - Programming File Generation Report
 - Generate PROM, ACE, or JTAG File
 - Configure Device (iMPACT)
 - Update Bitstream with Processor Data

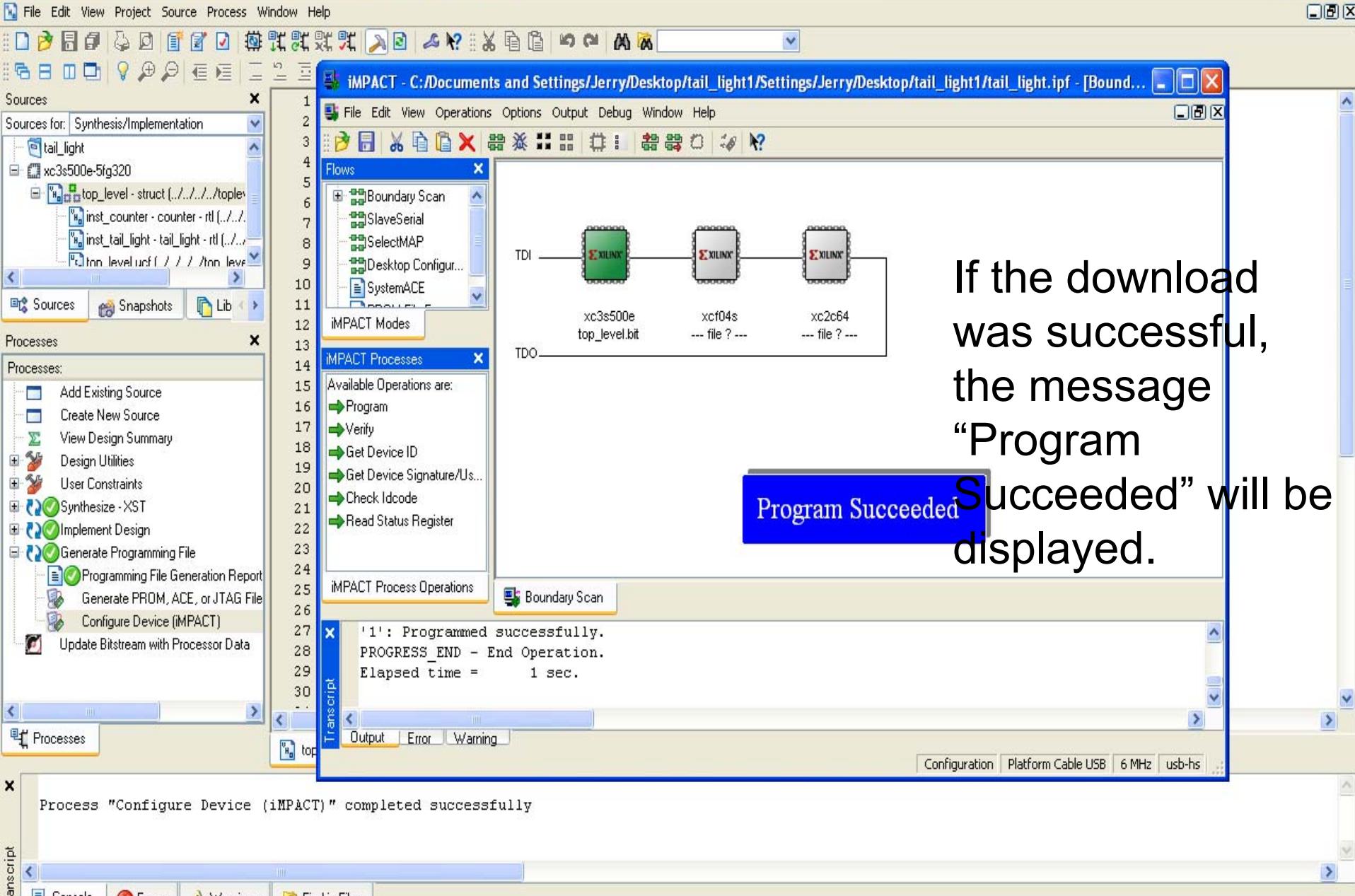


Process "Configure Device (iMPACT)"



Ready

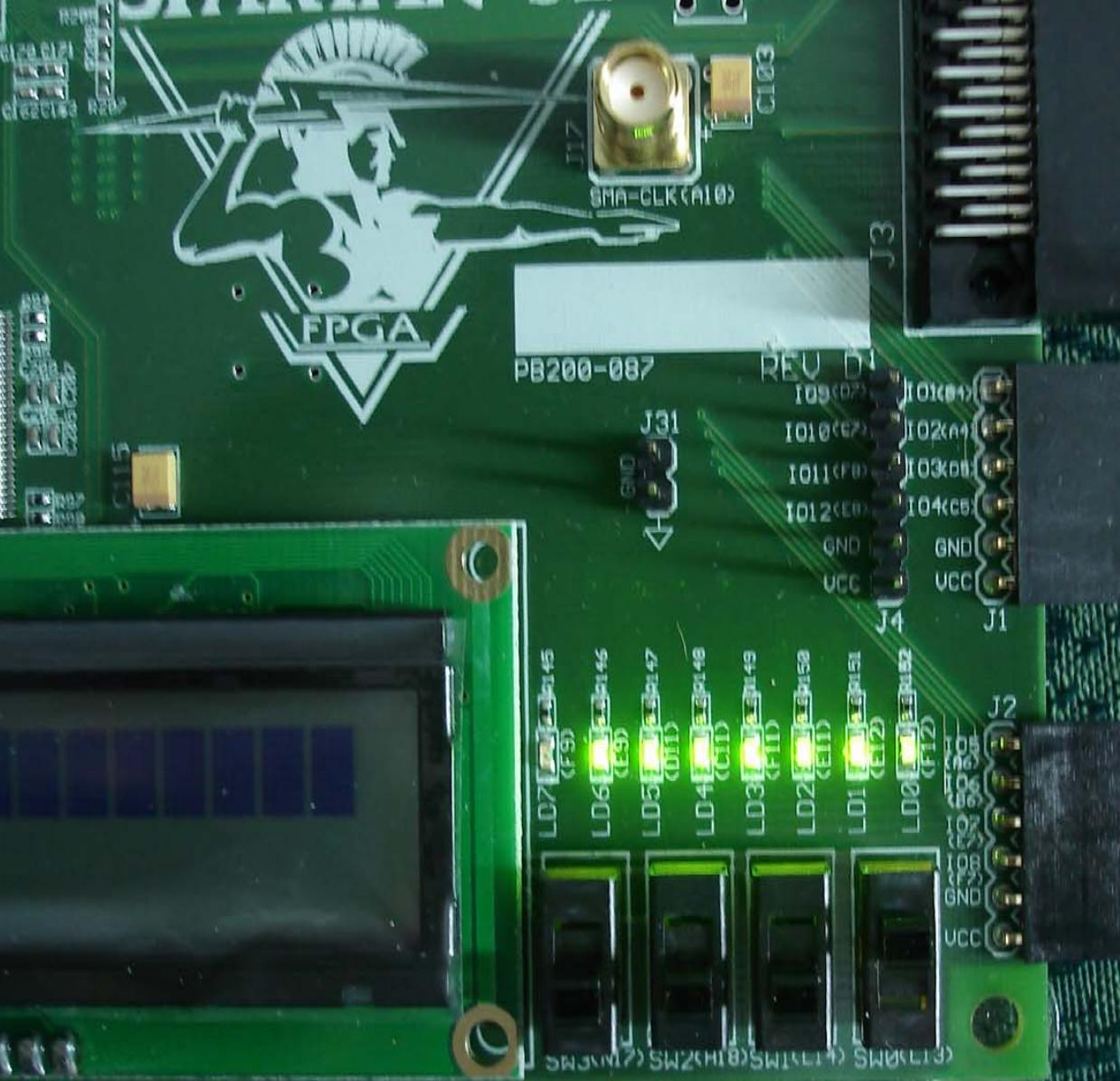




If the download was successful, the message “Program Succeeded” will be displayed.

Program Succeeded

Intel®
Flash
Memory



FPGA Demo



Intel®
Flash
Memory

