

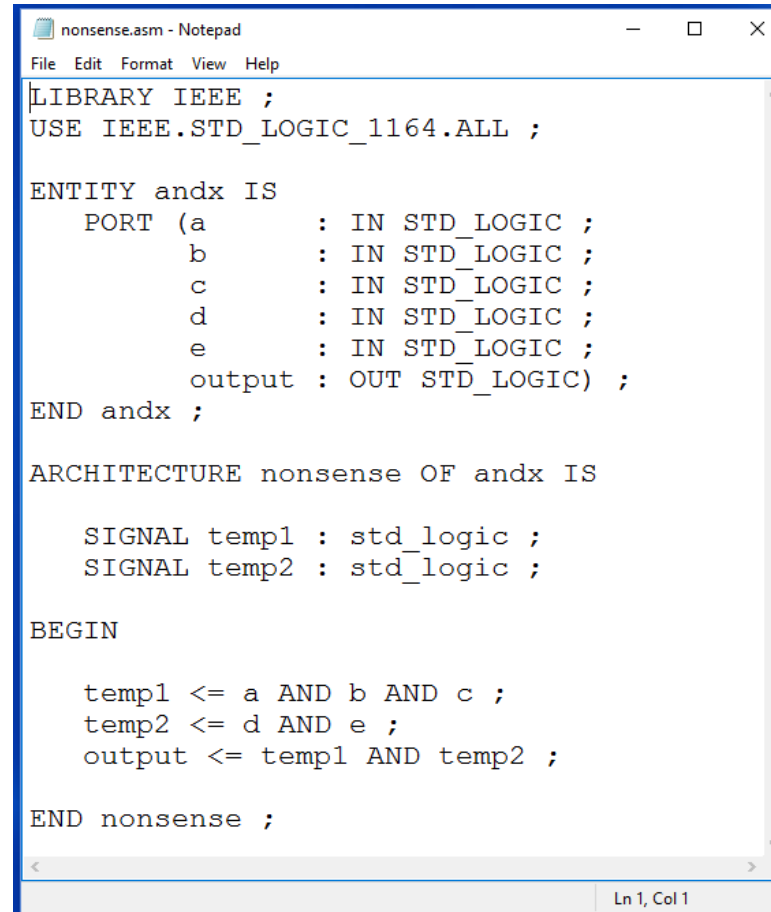
Vivado V2017.2 Tutorial

William D. Richard, Ph.D.

September 19, 2017

Updated September 10, 2018

Create your source using a text editor: example.vhd



```
nonsense.asm - Notepad
File Edit Format View Help
LIBRARY IEEE ;
USE IEEE.STD_LOGIC_1164.ALL ;

ENTITY andx IS
    PORT (a      : IN STD_LOGIC ;
          b      : IN STD_LOGIC ;
          c      : IN STD_LOGIC ;
          d      : IN STD_LOGIC ;
          e      : IN STD_LOGIC ;
          output : OUT STD_LOGIC) ;
END andx ;

ARCHITECTURE nonsense OF andx IS

    SIGNAL temp1 : std_logic ;
    SIGNAL temp2 : std_logic ;

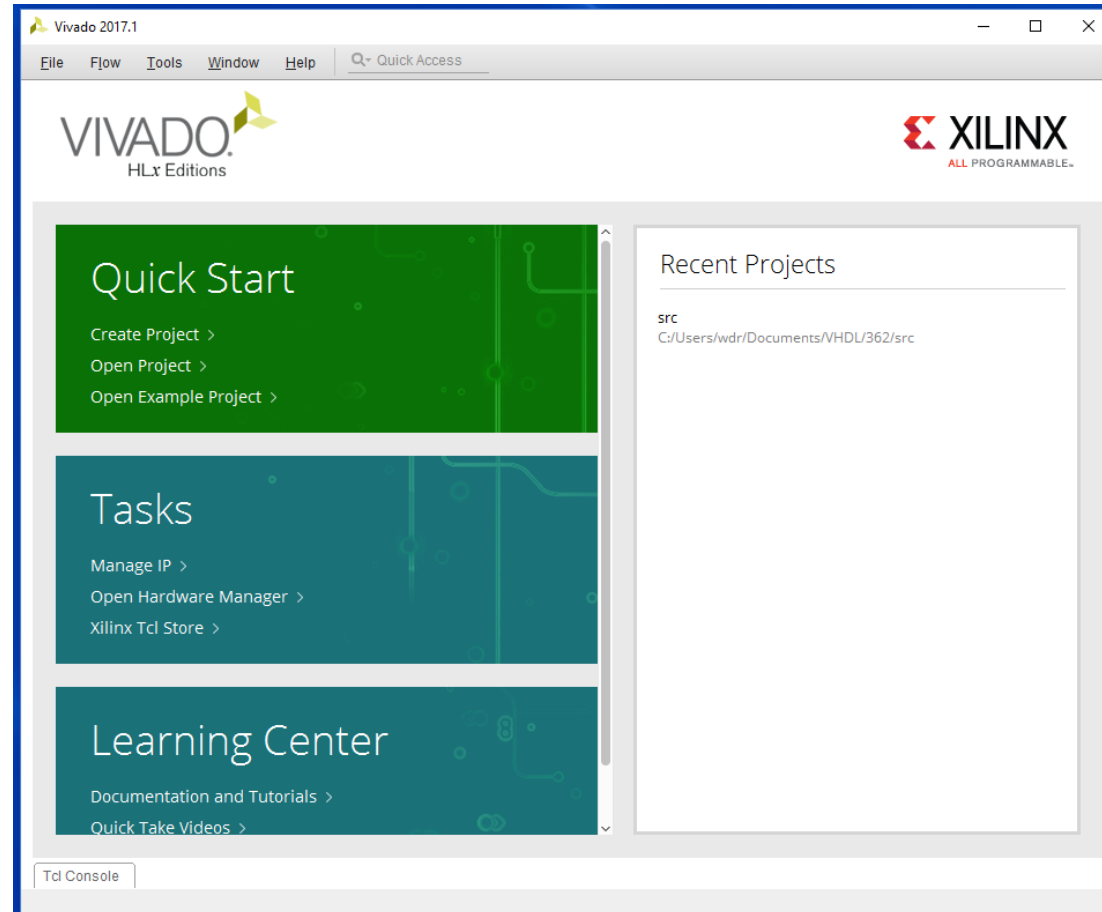
BEGIN

    temp1 <= a AND b AND c ;
    temp2 <= d AND e ;
    output <= temp1 AND temp2 ;

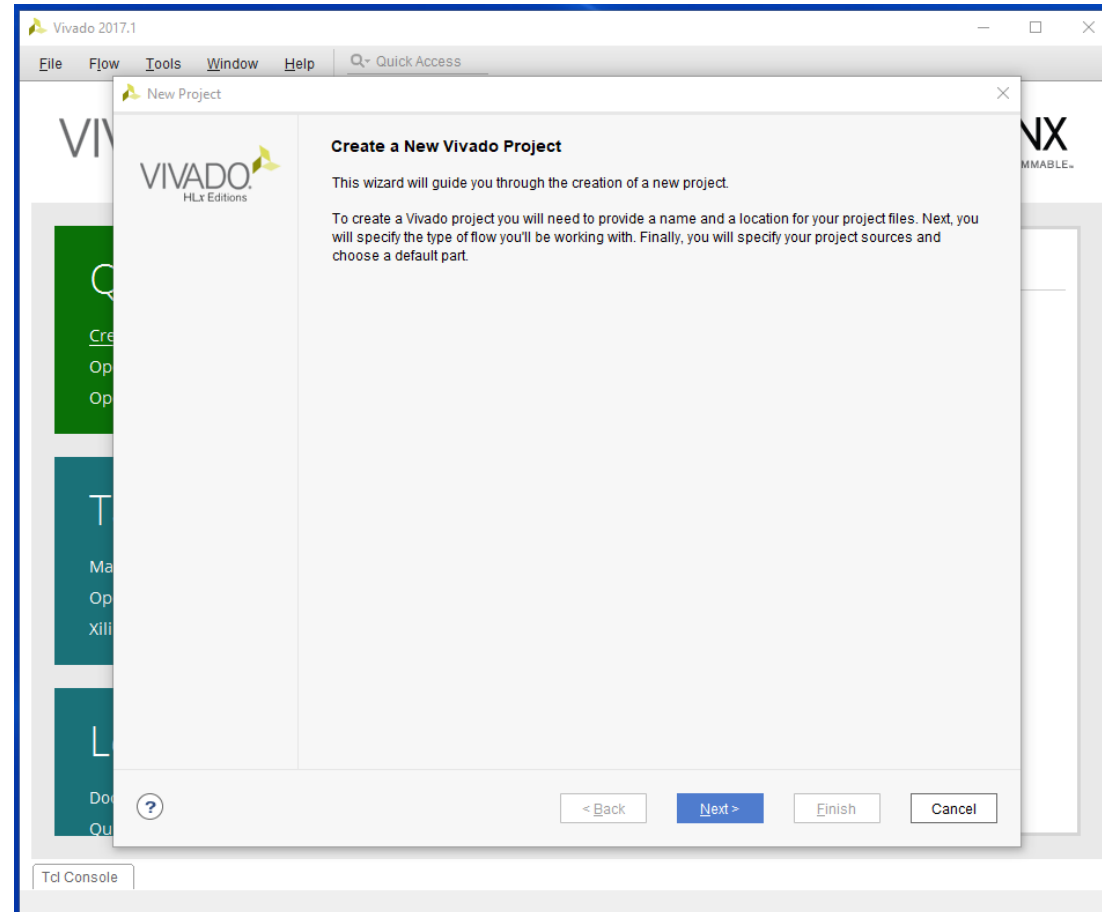
END nonsense ;

Ln 1, Col 1
```

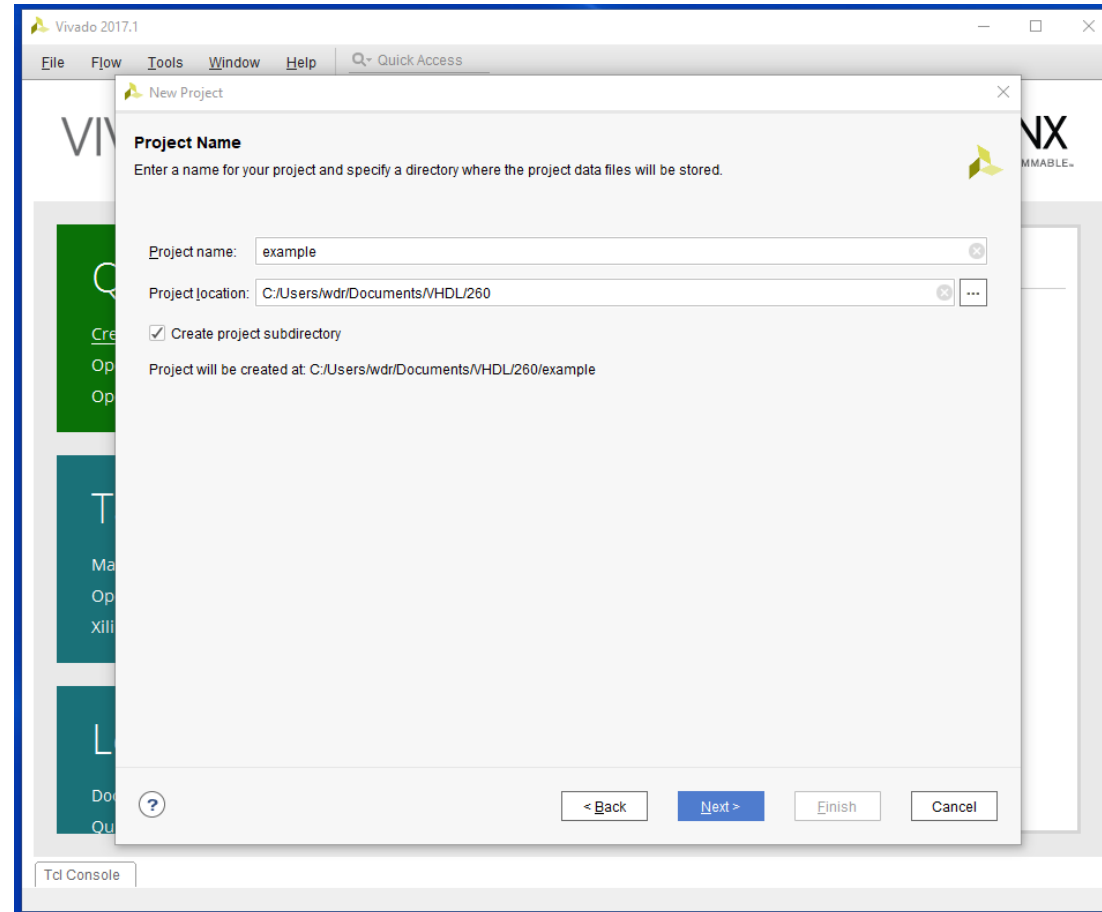
Run Vivado and select “Create New Project”



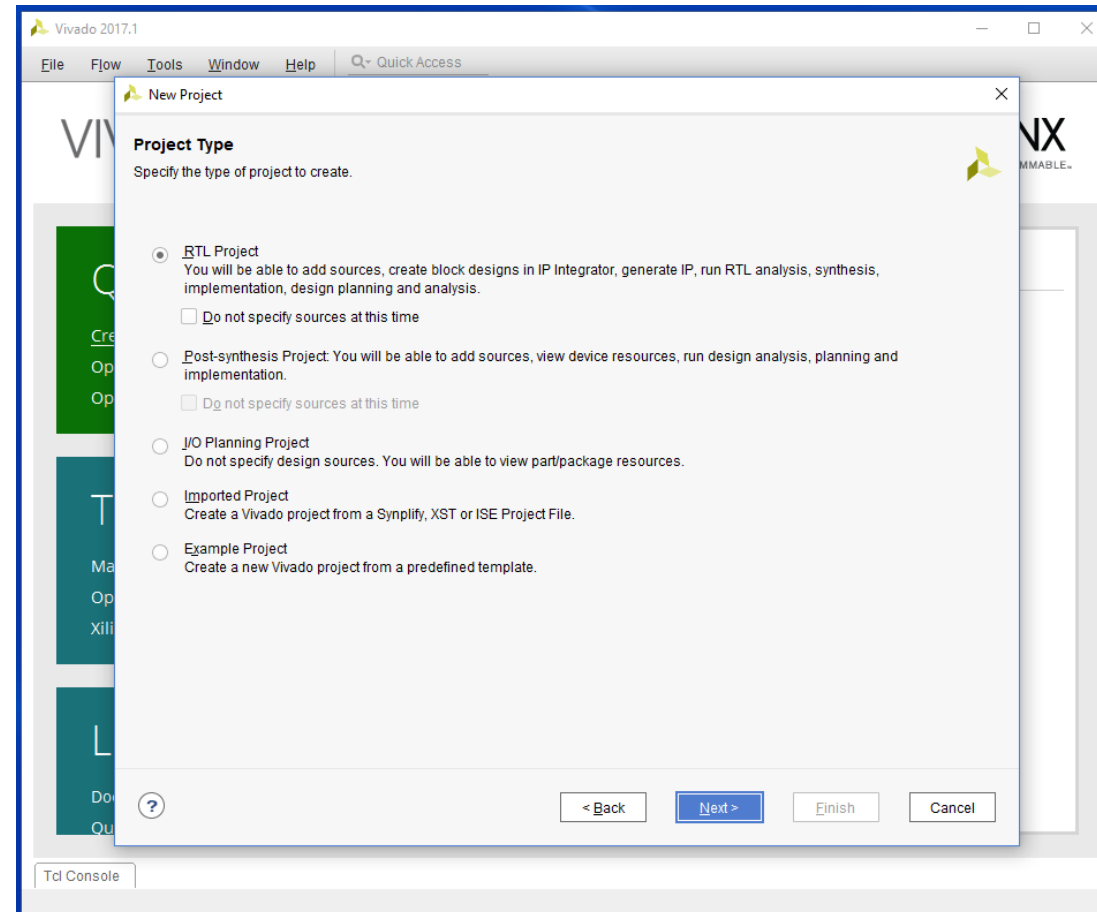
Read the info screen and click “Next”



Select the project directory and name your project



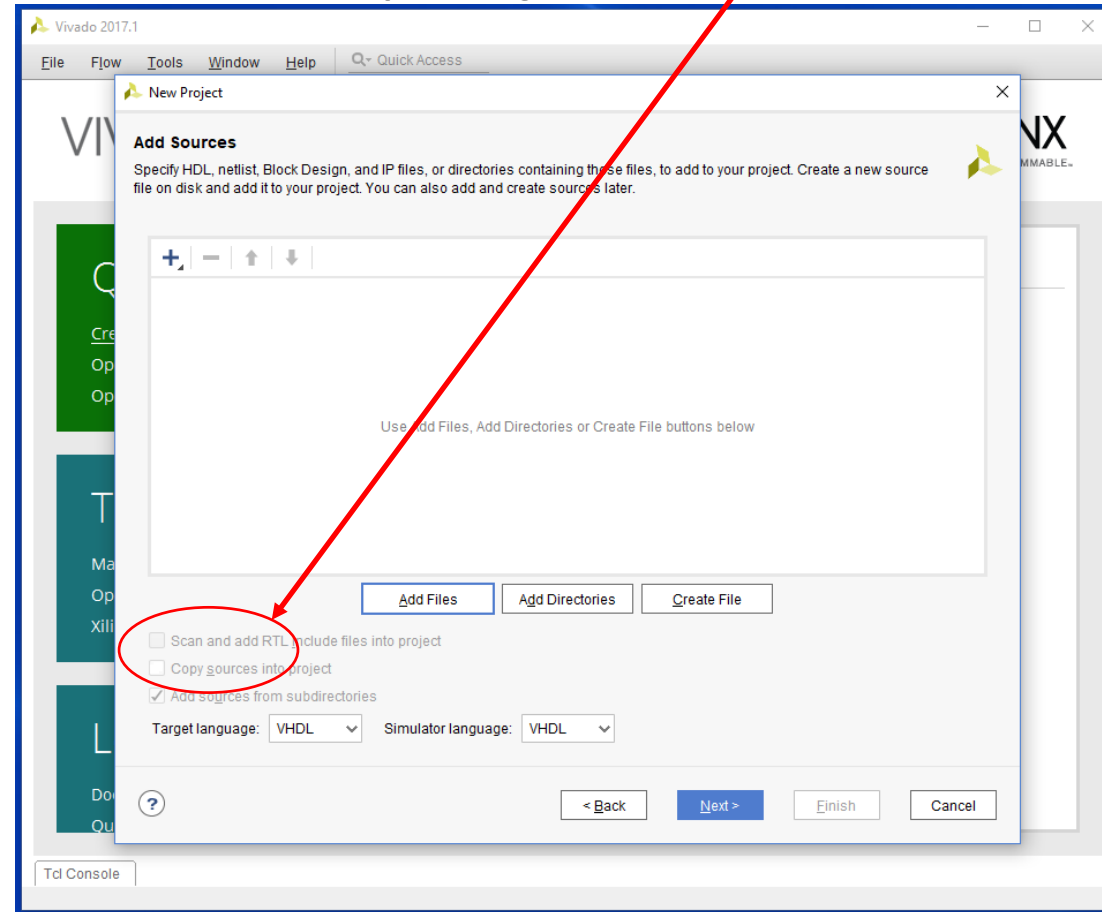
Select “RTL Project” and click “Next”



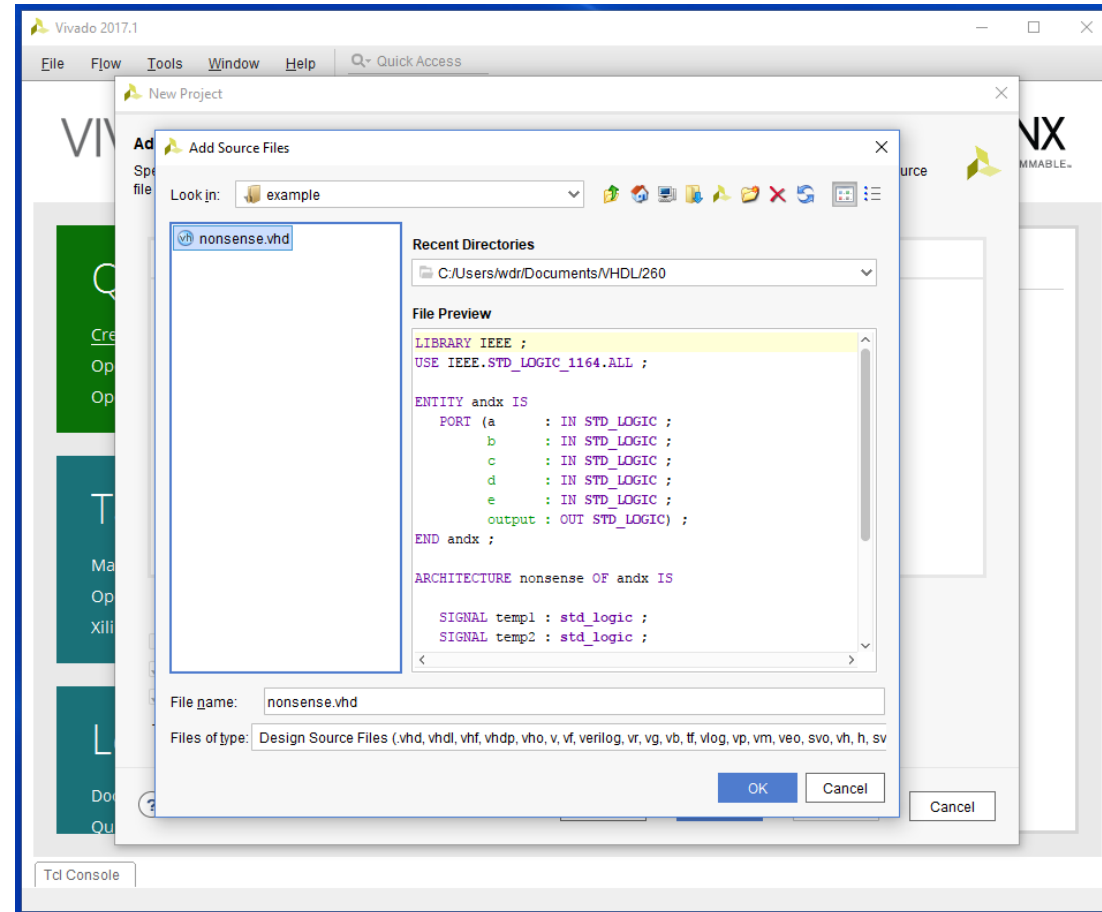
This is a very important step!

- After you create your project by clicking “next” on the last screen, your root project subdirectory is created
- Copy your source files into the root directory at this point, always keeping a copy of any work you did in case you mess up things as you start learning Vivado
- On the next screen, if you check “Copy sources into project, they will be automatically stuck in a subdirectory below the root project directory, and when you edit your sources in Vivado you will not be editing the files in the root directory – you will probably become very confused!

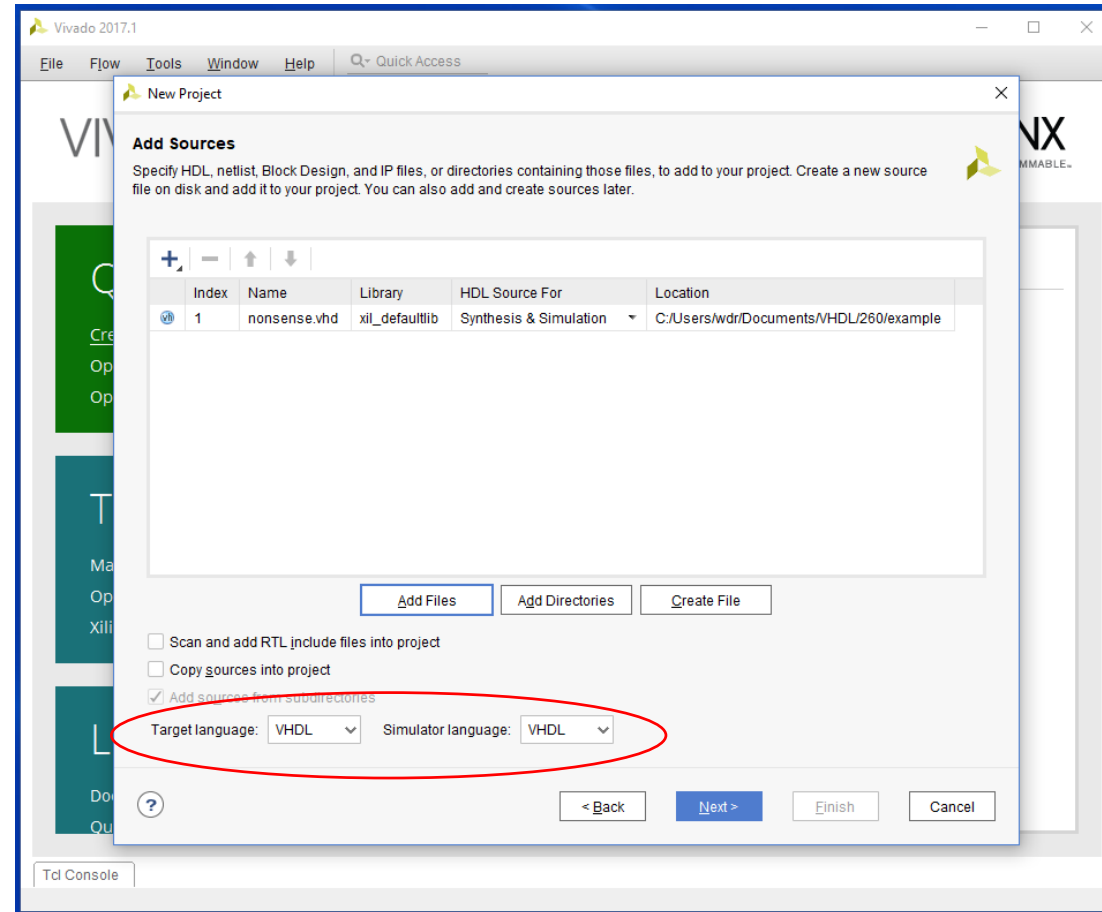
Select “Add Files” from the “Add Sources” GUI:
Make sure you don’t check “Copy sources into project!”



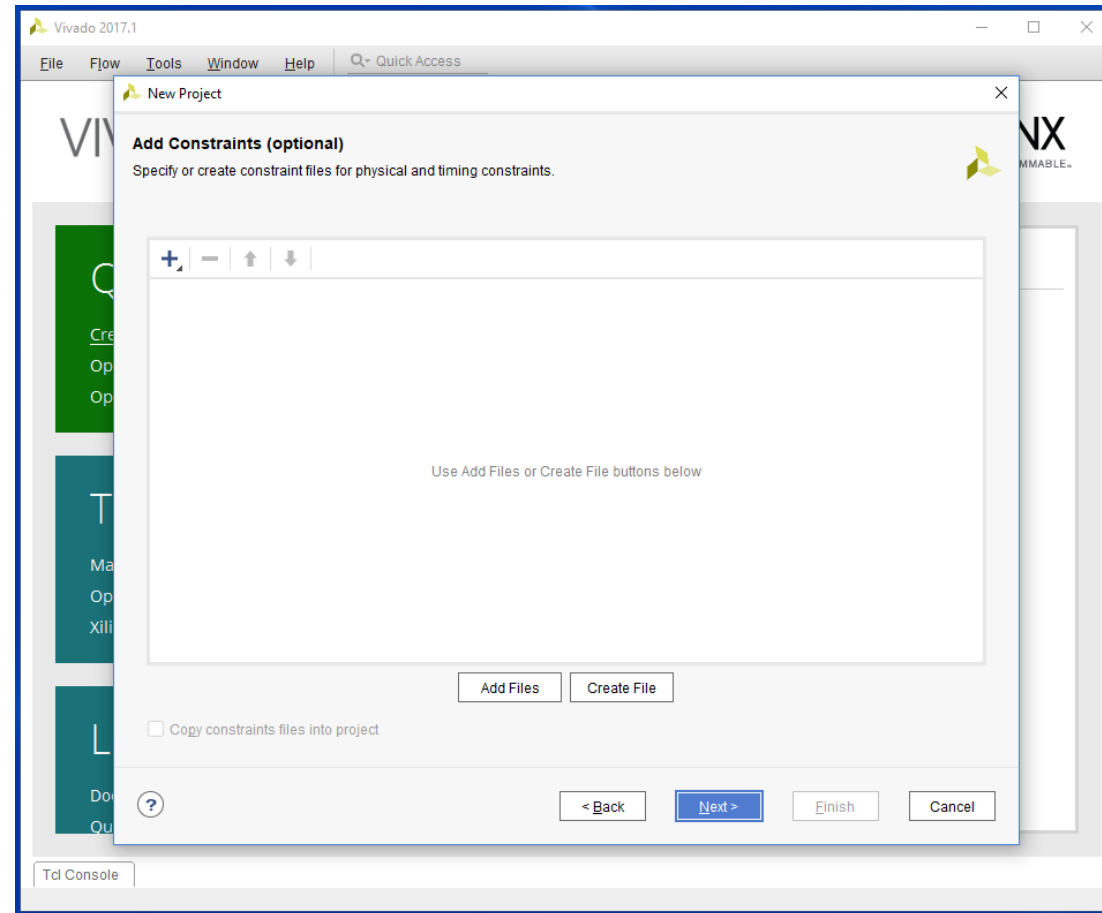
Navigate to your project directory and select your source example.vhd file, then click “OK”



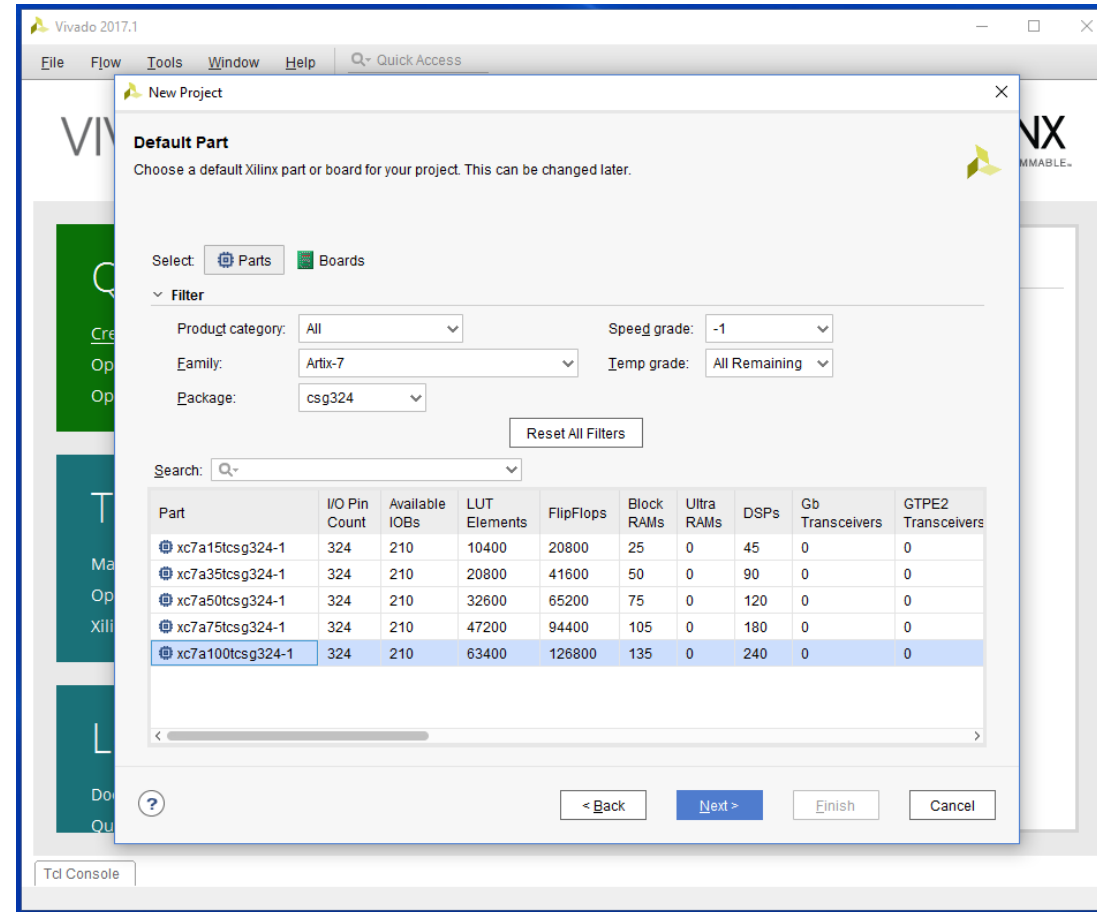
Make sure the Target and Simulator Language is set to VHDL, then click “Next”



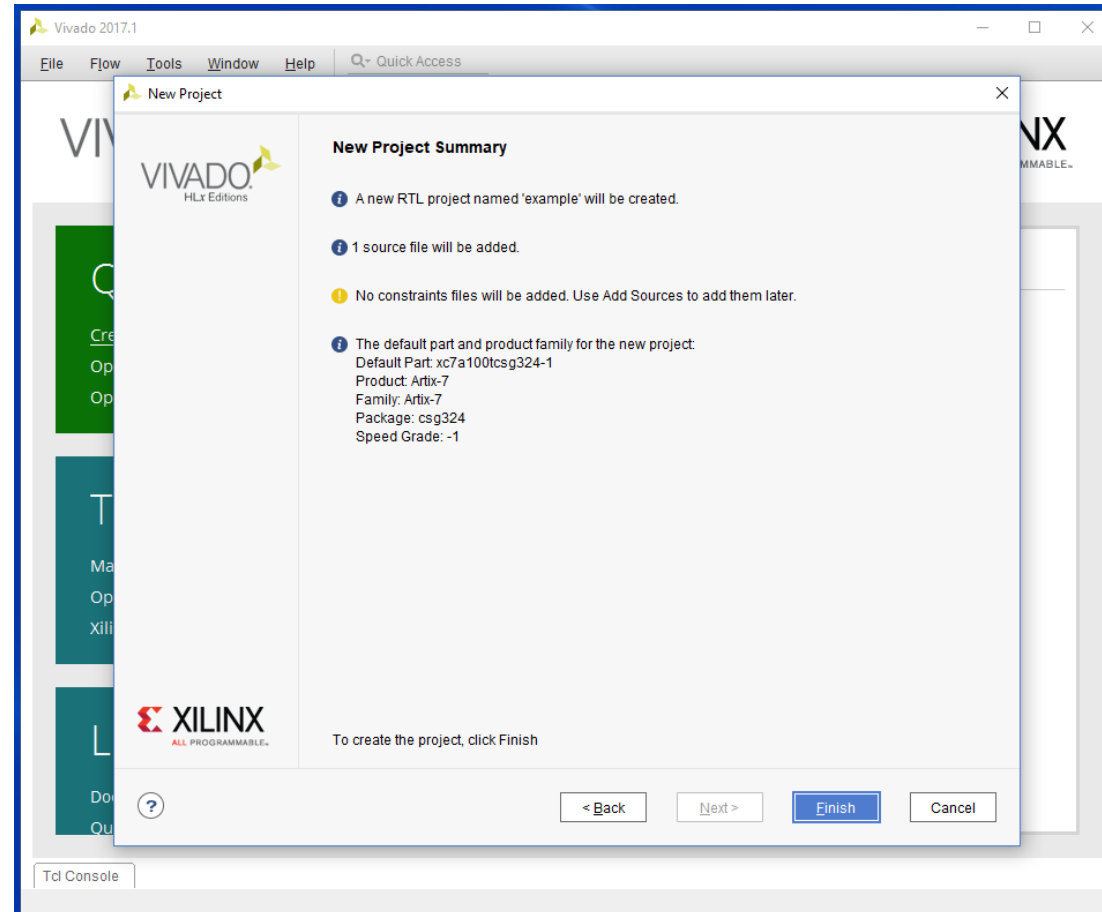
Don't add add constraints now, just select "Next"



Select the FPGA used on the CSE 260M demo board (Xilinx Artix 7 xc7a100tcsfg324-1)



Read the info screen and click “Finish”



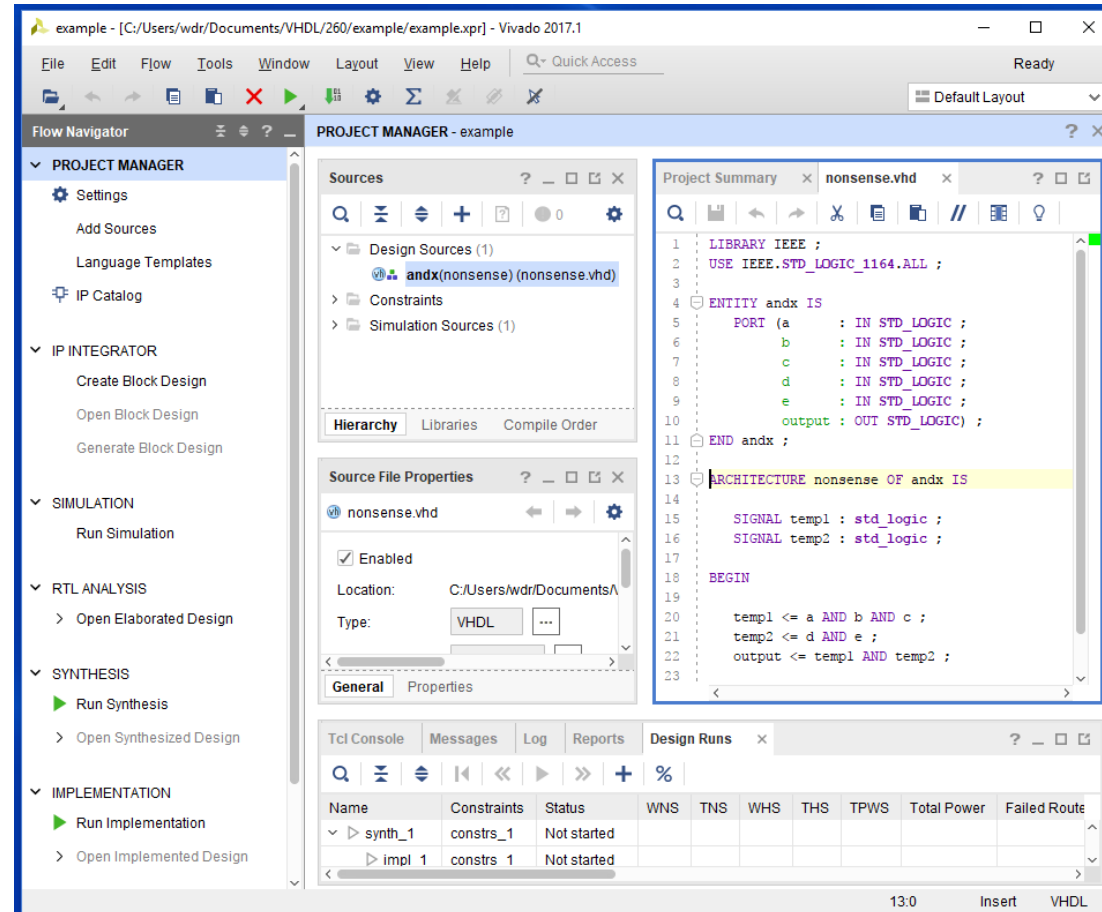
This is what you should get: A Vivado Project!

The screenshot displays the Vivado 2017.1 Project Manager interface for a project named 'example'. The window title is 'example - [C:/Users/wdr/Documents/VHDL/260/example/example.xpr] - Vivado 2017.1'. The interface is divided into several panes:

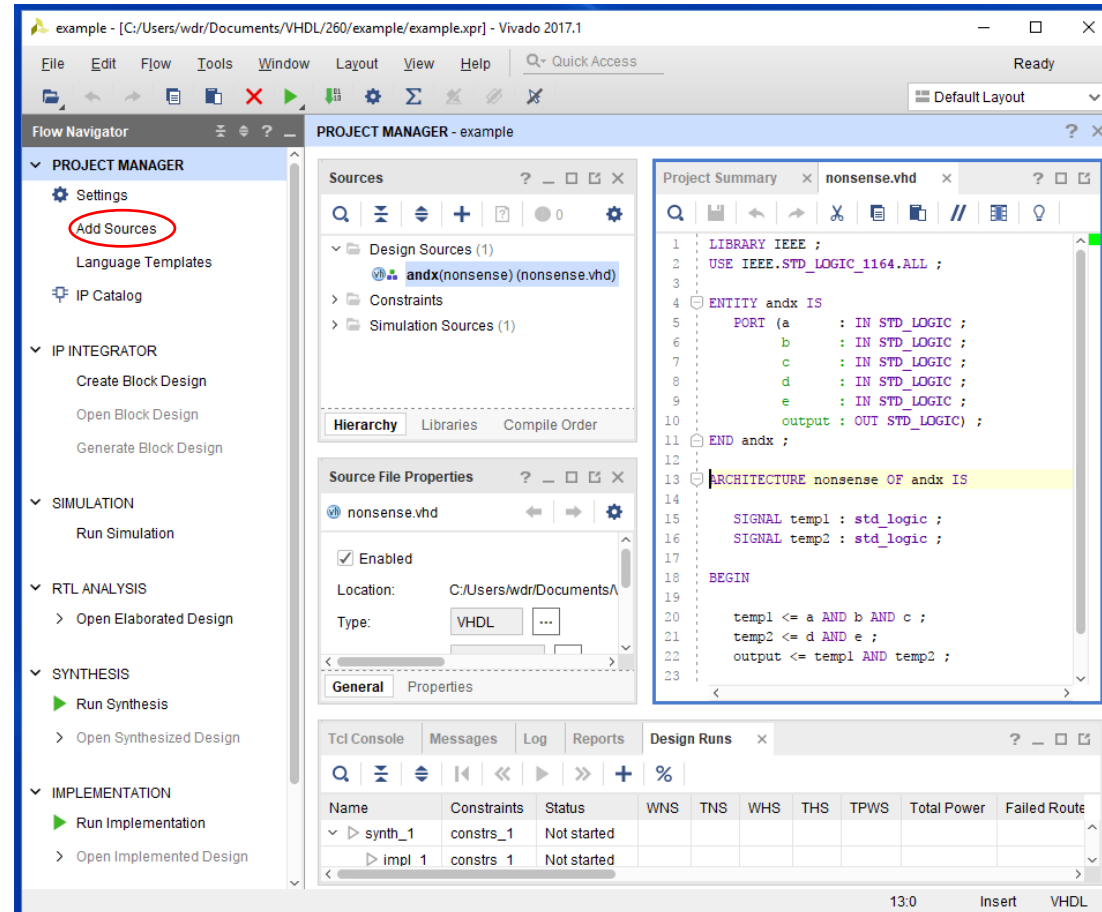
- Flow Navigator:** A vertical sidebar on the left containing project management tasks such as 'Settings', 'Add Sources', 'Language Templates', 'IP Catalog', 'Create Block Design', 'Open Block Design', 'Generate Block Design', 'Run Simulation', 'Open Elaborated Design', 'Run Synthesis', 'Open Synthesized Design', 'Run Implementation', and 'Open Implemented Design'.
- PROJECT MANAGER - example:** The main workspace, which includes:
 - Sources:** A tree view showing 'Design Sources (1)' containing 'andx(nonsense) (nonsense.vhd)', 'Constraints', and 'Simulation Sources (1)'. Below this are tabs for 'Hierarchy', 'Libraries', and 'Compile Order'.
 - Properties:** A pane for viewing object properties, currently showing 'Select an object to see properties'.
 - Project Summary:** A pane on the right displaying project details:
 - Project name: example
 - Project location: C:/Users/wdr/Doc
 - Product family: Artix-7
 - Project part: xc7a100tcsq324-
 - Top module name: andx
 - Target language: VHDL
 - Simulator language: VHDL
 - Synthesis Status: Not started
- Design Runs:** A table at the bottom showing the status of synthesis runs.

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes
synth_1	constrs_1	Not started							
impl_1	constrs_1	Not started							

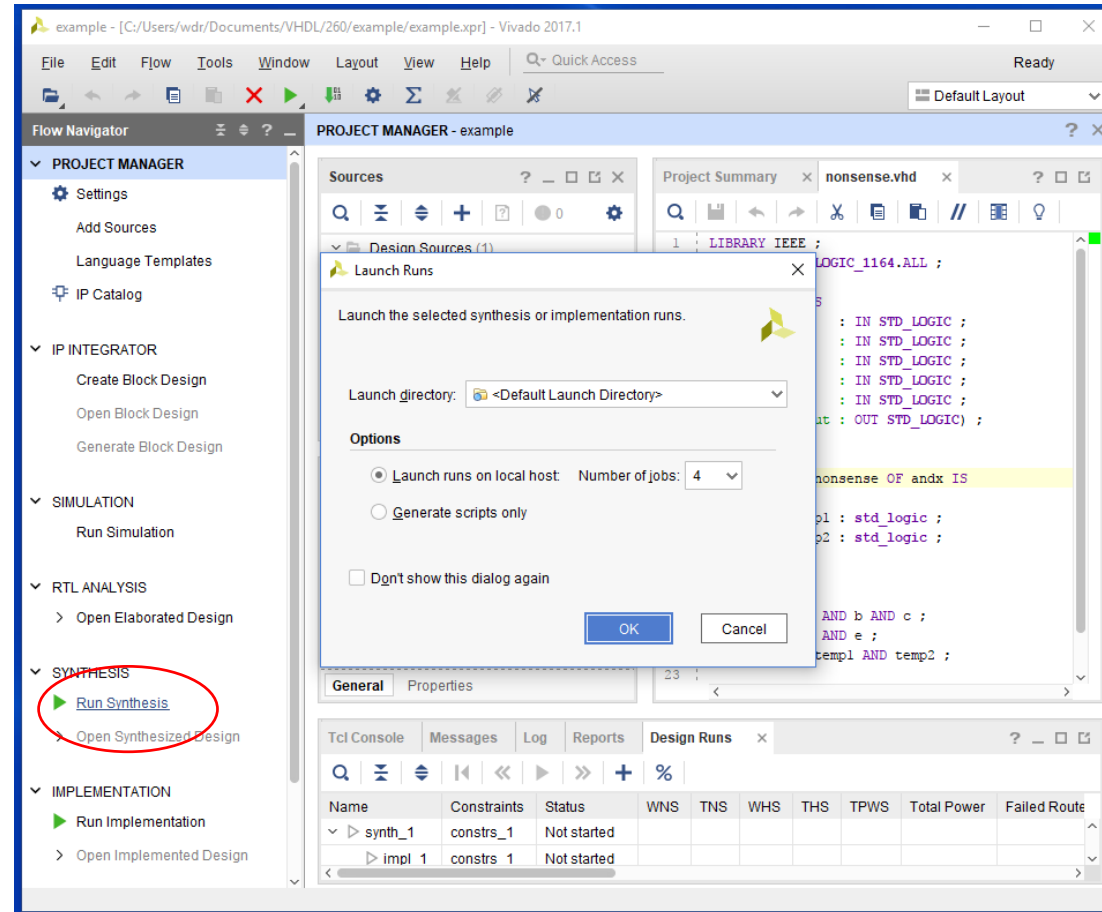
The Vivado Project Manager allows you to view and edit your source file



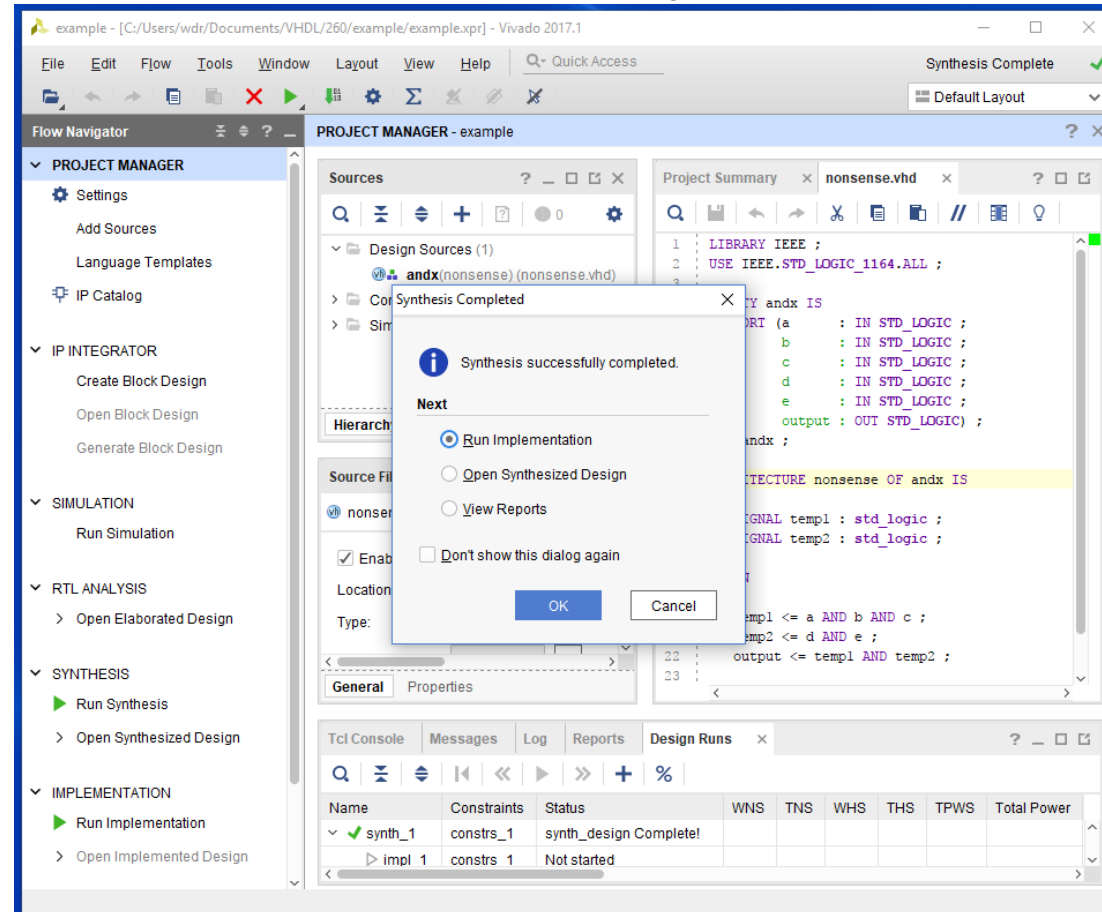
If you didn't create your VHDL source file ahead of time, you can use the GUI to create one



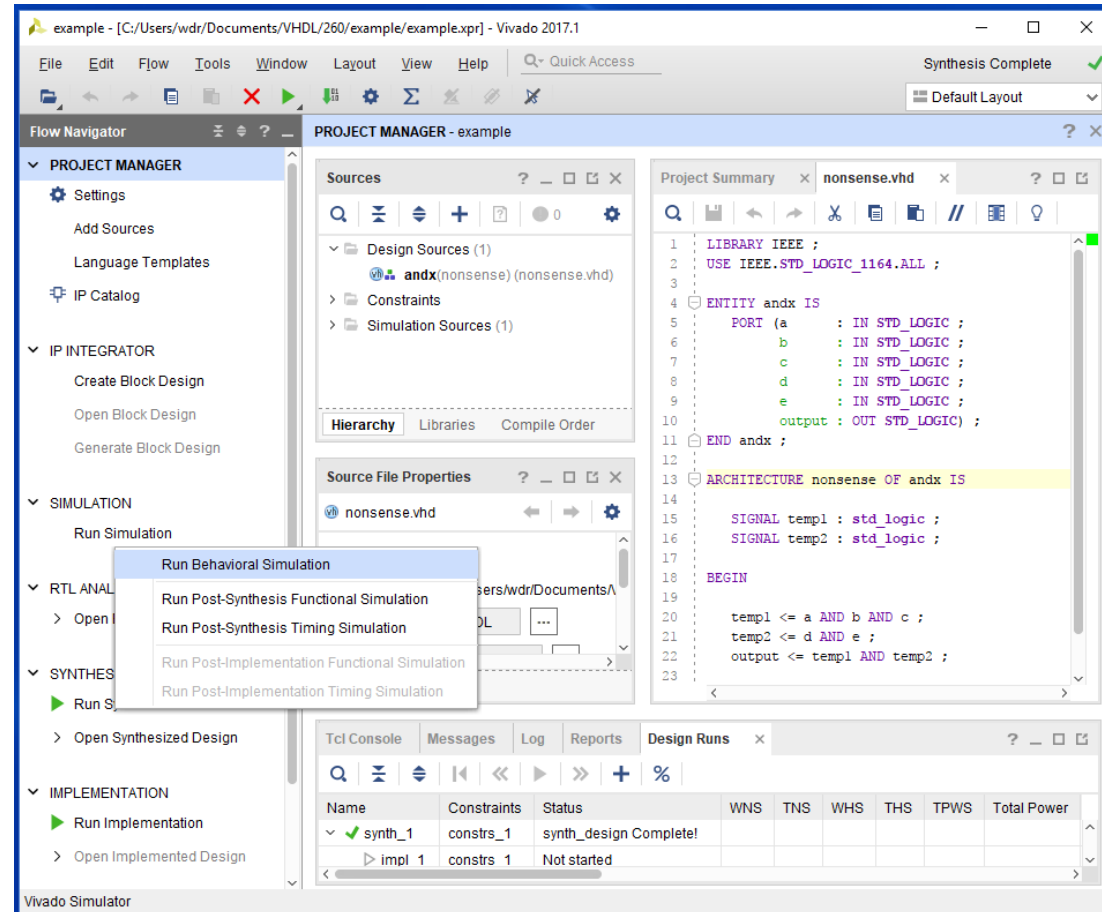
You have to synthesize your design before you can simulate it!



If you are just trying to simulate your design, you can select “Cancel” once you get a successful synthesis: No need to implement the design!



Run the simulator by right clicking on the “Run Simulation” menu item



The simulator will take a few seconds to initialize (minutes for big designs)

The screenshot displays the Vivado 2017.1 software interface during a behavioral simulation. The main window is titled "SIMULATION - Behavioral Simulation - Functional - sim_1 - andx". The interface is divided into several panes:

- Flow Navigator:** Shows the project structure with sections for PROJECT MANAGER, IP INTEGRATOR, SIMULATION, RTL ANALYSIS, SYNTHESIS, and IMPLEMENTATION. The "SIMULATION" section is currently active, showing "Run Simulation".
- Objects:** A table listing the objects in the simulation. The table has columns for Name and Value.
- Waveform:** A timing diagram showing signals over time. The time axis is labeled "1,000,000 ps". The signals listed are a, b, c, d, e, output, temp1, and temp2. The values for these signals are shown as "U" (Unknown) in the current view.
- Tcl Console:** Displays the simulation log. The log shows the following messages:

```
# run 1000ns
INFO: [USF-XSim-96] XSim completed. Design snapshot 'andx_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:04 ; elapsed = 00:00:10 . Memory (MB): peak = 940.156 ; gain = 0.000
```

The status bar at the bottom right indicates "Sim Time: 1 us".

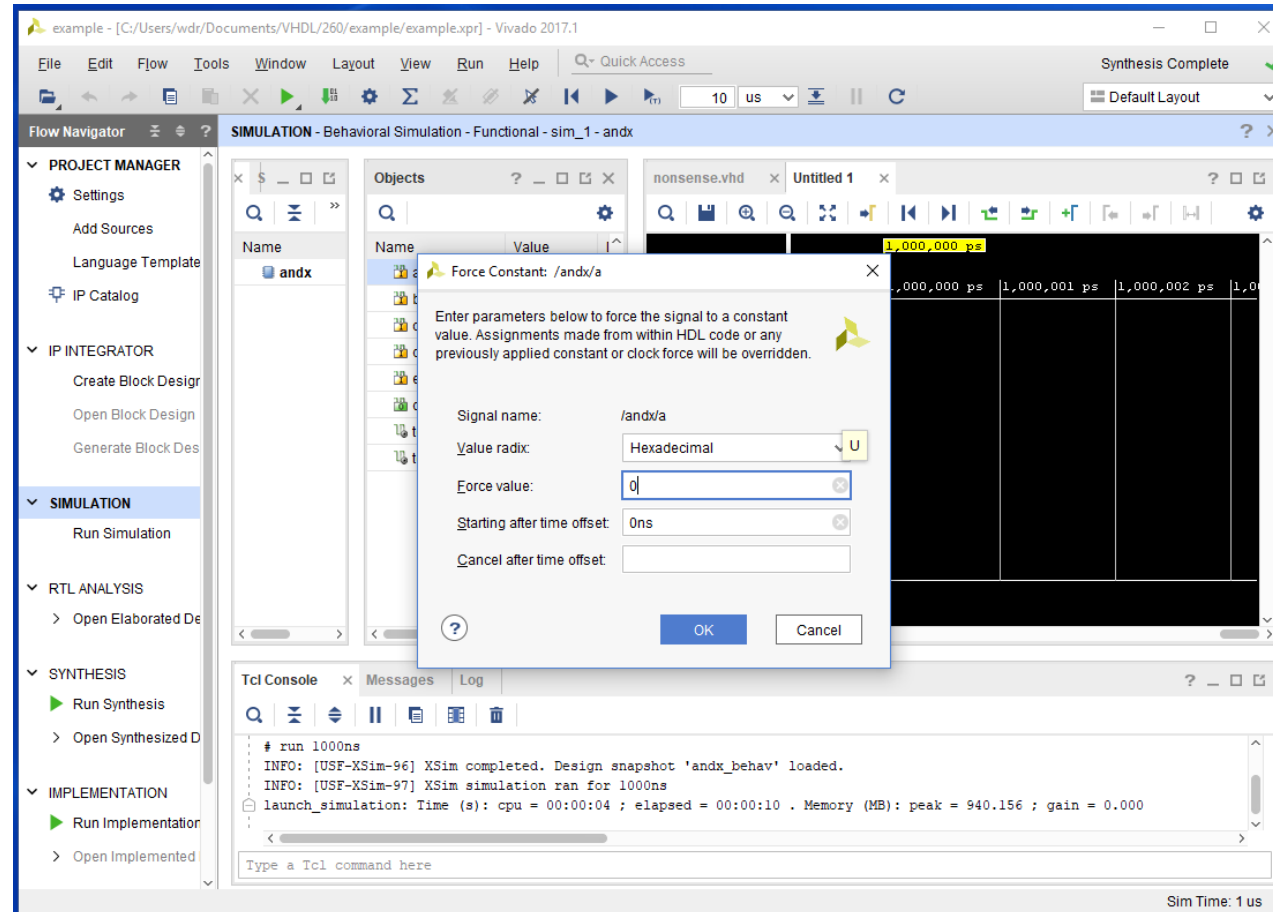
Force the initial stimulus values by right clicking on each input in the “Objects” menu

The screenshot displays the Vivado 2017.1 software interface during a behavioral simulation. The main window shows a simulation of a circuit with several input signals (a, b, c, d, e) and two temperature signals (temp1, temp2). The 'Objects' window is open, showing a list of these signals. A context menu is open over the signal 'a', with the 'Force Constant...' option selected. The menu also includes options like 'Go To Source Code', 'Show in Object Window', 'Report Drivers', 'Force Clock...', 'Remove Force', and standard editing functions like 'Cut', 'Copy', 'Paste', and 'Delete'. The 'Tcl Console' at the bottom shows the simulation results, including the command '# run 1000ns' and the completion of the simulation.

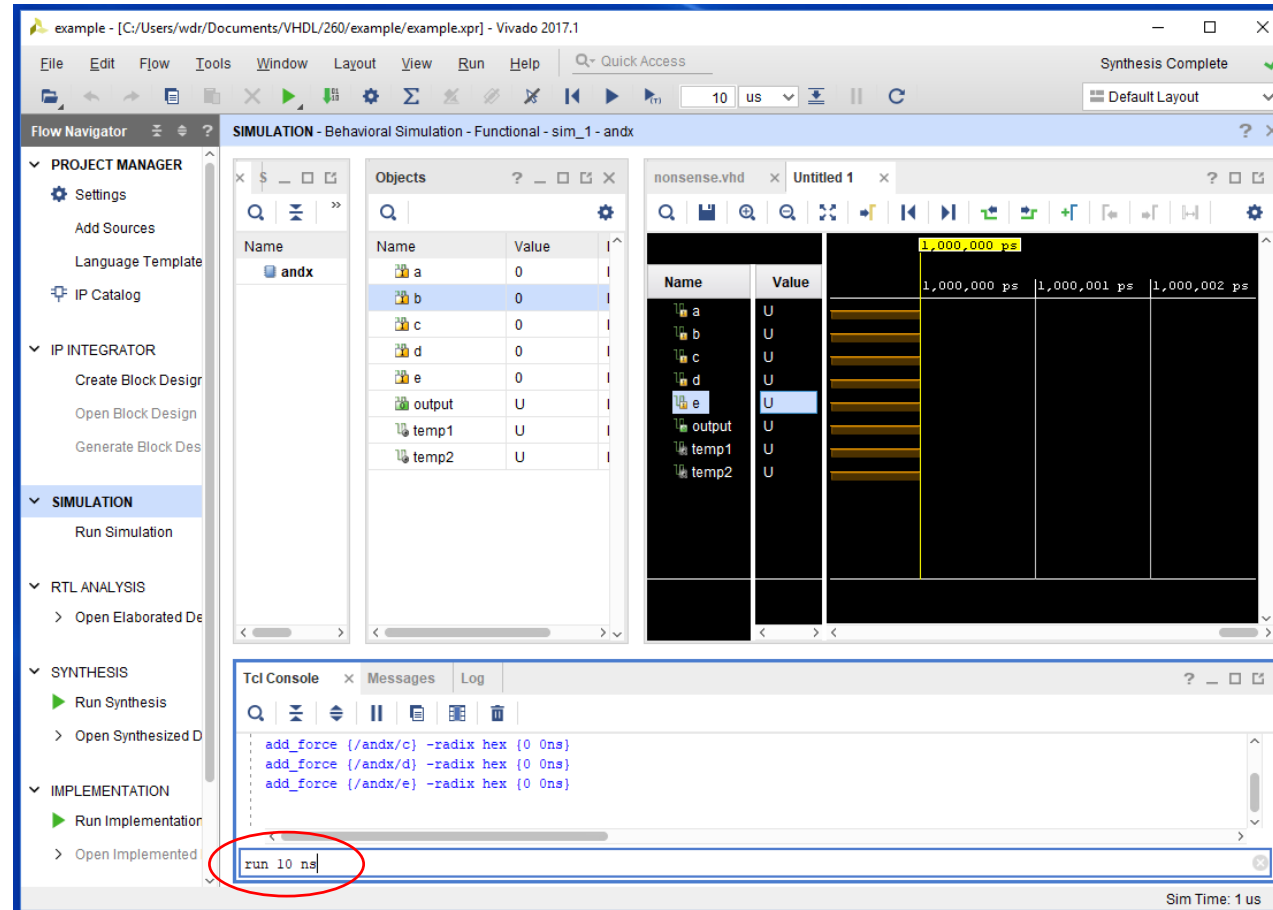
Name	Value
a	U
b	U
c	U
d	U
e	U
output	U
temp1	U
temp2	U

```
# run 1000ns
INFO: [USF-XSim-96] XSim completed. Design snapshot
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:04 ; elapsed
```

For this example, signal “a” is forced to logic 0



Run the simulator for 10 ns by typing “Run 10 ns” in the TCL Console



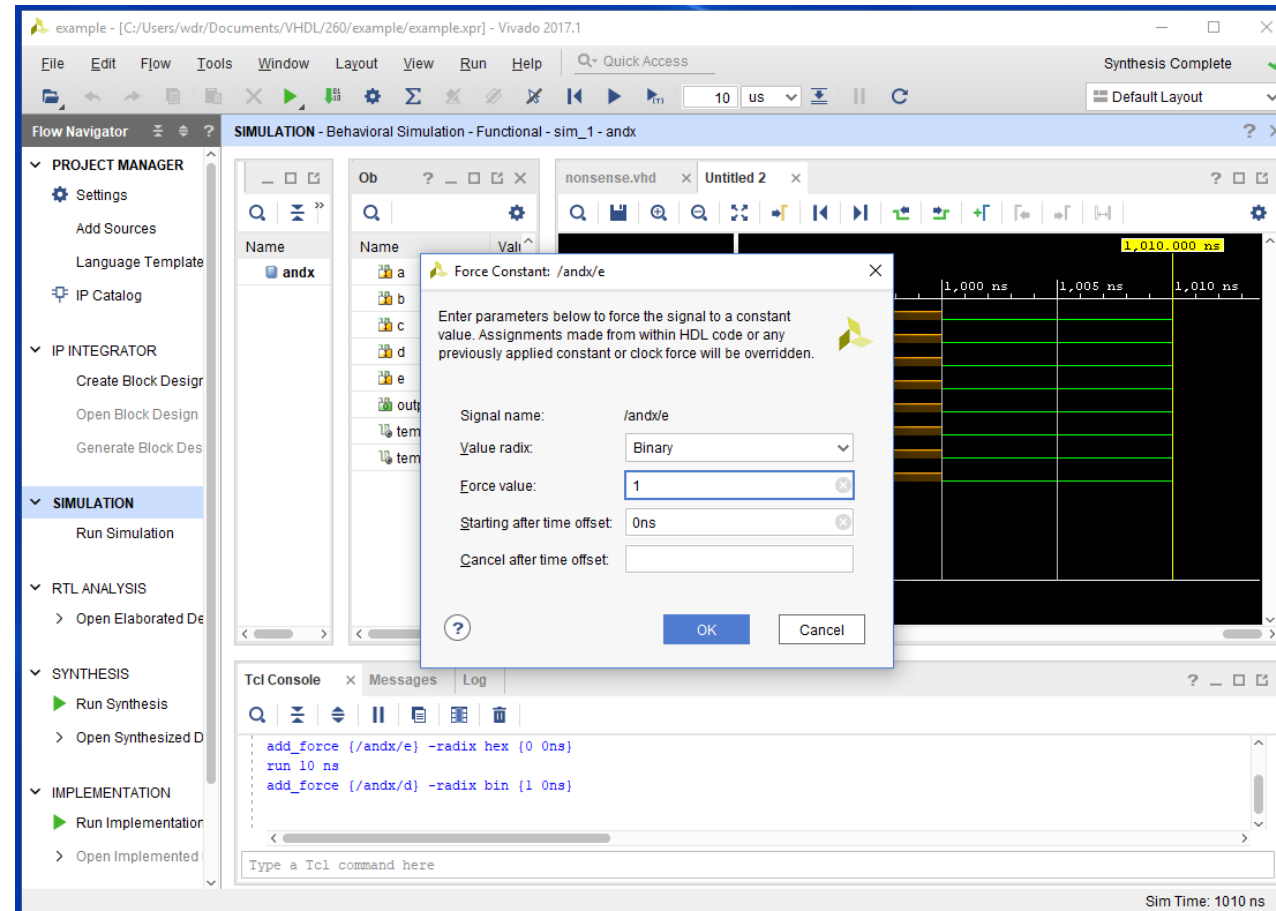
Use the zoom “magnifying glass” controls to adjust the simulation display

The screenshot shows the Vivado 2017.1 simulation environment. The main window displays a behavioral simulation of a design named 'andx'. The simulation is running at 10 ns. The waveform display shows signals 'a', 'b', 'c', 'd', 'e', 'output', 'temp1', and 'temp2'. The signal 'e' is highlighted, and its value is shown as 0. The waveform is zoomed in, with a magnifying glass icon visible in the top right corner of the waveform area. The Tcl Console at the bottom shows the following commands:

```
add_force [/andx/d] -radix hex {0 0ns}
add_force [/andx/e] -radix hex {0 0ns}
run 10 ns
```

The status bar at the bottom right indicates 'Sim Time: 1010 ns'.

Force additional stimulus values by right clicking on inputs in the “Objects” menu



Run the simulator for an additional 10 ns after changing stimulus values using the TCL Console

The screenshot displays the Vivado 2017.1 interface during a behavioral simulation. The Tcl Console at the bottom shows the following commands:

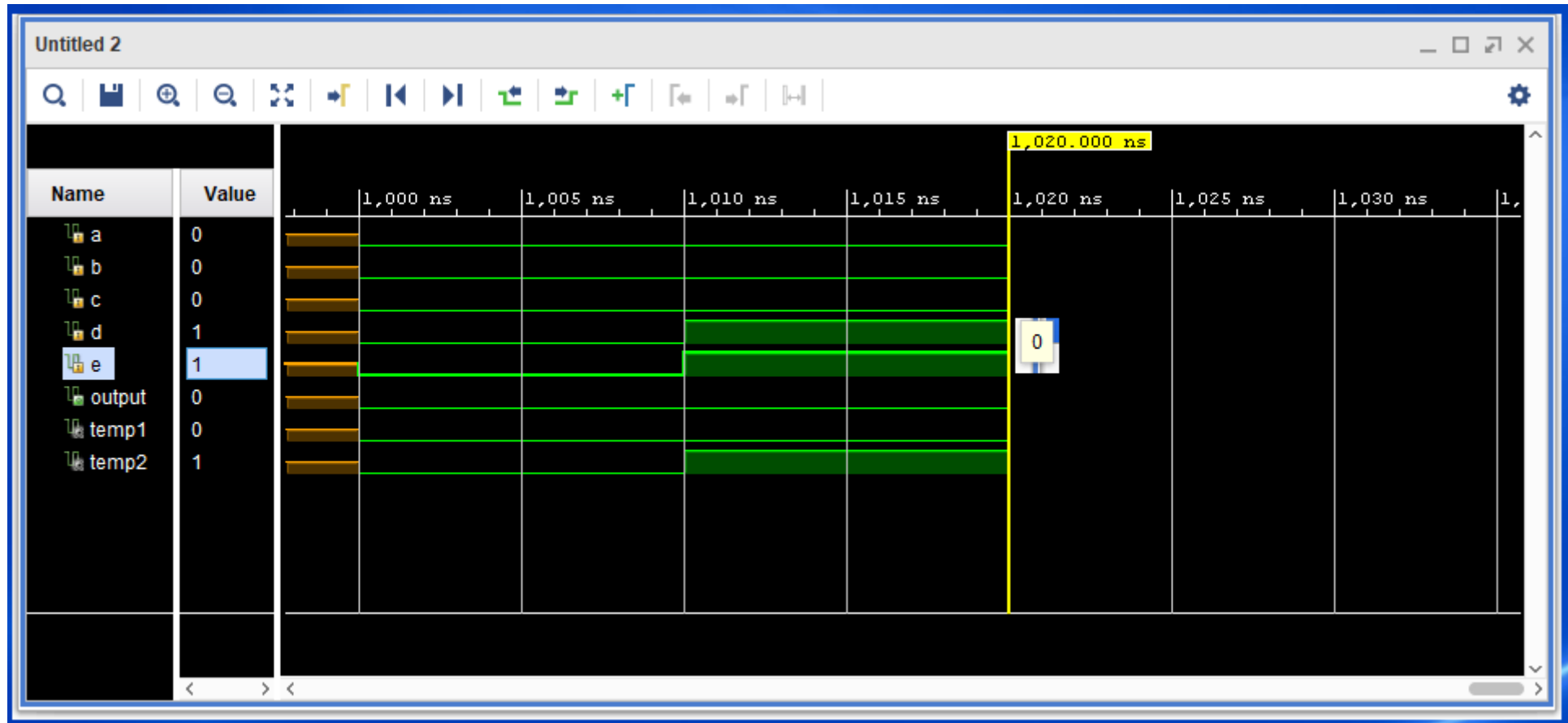
```
add_force {/andx/d} -radix bin (1 0ns)
add_force {/andx/e} -radix bin (1 0ns)
run 10 ns
```

The waveform viewer shows the simulation results for signals a, b, c, d, e, output, temp1, and temp2. The time axis is marked at 1,000 ns, 1,005 ns, 1,010 ns, and 1,015 ns. A yellow vertical line indicates the current simulation time at 1,020.000 ns. The signal values are as follows:

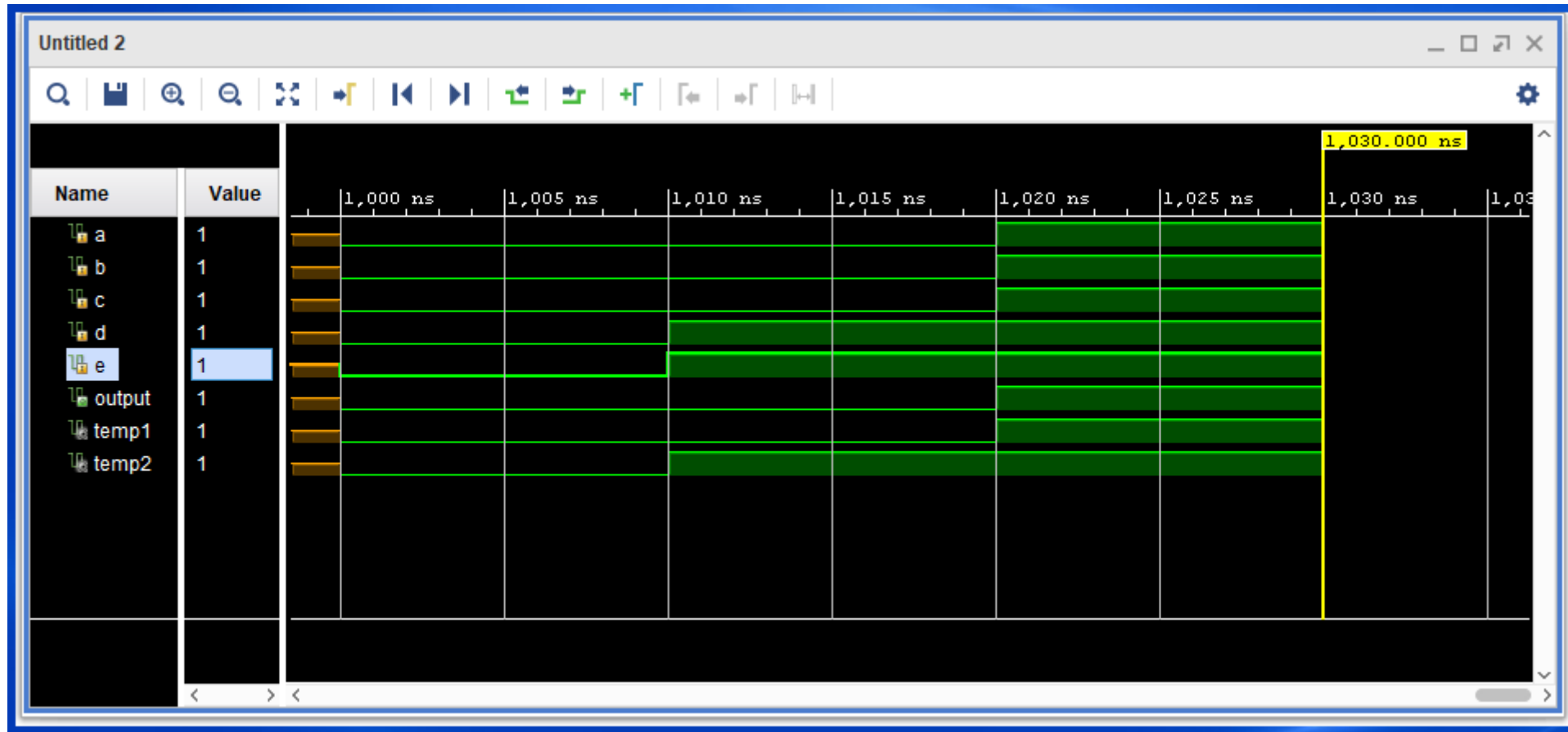
Name	Value
a	0
b	0
c	0
d	1
e	1
output	0
temp1	0
temp2	1

The status bar at the bottom right indicates the simulation time: Sim Time: 1020 ns.

You can unlock the simulation window from the Vivado GUI



If you force all inputs to '1', the output will go high after you run an additional 10 ns!



Open the Elaborated Design produced by the Synthesizer to see what your VHDL implied!

The screenshot shows the Vivado 2017.1 Project Manager interface. The 'PROJECT MANAGER - example' window is active, displaying the 'Sources' tab with a tree view of design sources. The 'Open Elaborated Design' option under the 'RTL ANALYSIS' section is circled in red. The 'Project Summary' window is also visible, showing project details and synthesis status.

Project Summary

Settings Edit

Project name: example
Project location: C:/Users/wdr/Documents/VHDL/260/example
Product family: Artix-7
Project part: xc7a100tcsq324-1
Top module name: andx
Target language: VHDL
Simulator language: VHDL

Synthesis Implement

Status: ✔ Complete Status:

Tcl Console Messages Log Reports Design Runs

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM
synth_1	constrs_1	synth_design Complete!								1	0	0.0
impl_1	constrs_1	Not started										

Analyze and constrain an elaborated netlist

Open the Synthesized Design and view the schematic: A single LUT!

The screenshot displays the Vivado 2017.2.1 interface for a synthesized design. The main window shows a schematic diagram of a single LUT5 cell. The LUT5 cell has five inputs labeled a, b, c, d, and e, each connected to an input buffer (IBUF). The LUT5 cell has two outputs labeled output_INST_0 and output_INST_0_I_1. The output_INST_0 is connected to an output buffer (OBUF), which is then connected to the final output pin. The I/O ports table at the bottom of the schematic window provides the following details:

Name	Direction	Interface	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	I
output	OUT				<input type="checkbox"/>		default (LVCMOS18)	1.800		
a	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
b	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
c	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
d	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
e	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		

Expand the Schematic so you can see the “6 I/O Ports” option; select it to view the pins

The screenshot shows the Vivado IDE interface for a synthesized design. The main window displays a schematic diagram of a circuit with a LUT5 and several buffers. The '6 I/O Ports' option is highlighted in the top right corner of the schematic window. Below the schematic, the 'Find Results' table lists the I/O ports and their properties.

Name	Direction	Interface	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	I
output	OUT				<input type="checkbox"/>		default (LVCMOS18)	1.800		
a	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
b	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
c	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
d	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		
e	IN				<input type="checkbox"/>		default (LVCMOS18)	1.800		

Assign the pins to SW(4:0) and LD(0) using the documentation for the demo board

The screenshot shows the Vivado IDE interface for a synthesized design. The top menu bar includes File, Edit, Flow, Tools, Window, Layout, View, and Help. The main window is divided into several panes:

- Flow Navigator:** Shows the project structure with 'SYNTHESIS' expanded, including options like 'Run Synthesis', 'Open Synthesized Design', and 'Report Timing Summary'.
- Project Summary:** Displays 'Device: nonsense.vhd' and 'Schematic'.
- Schematic:** Shows a logic diagram with five input buffers (a_IBUF_inst to e_IBUF_inst) connected to a LUT5, which is then connected to an output buffer (output_INST_0).
- I/O Ports:** A table listing the I/O ports and their physical pin assignments. The 'Package Pin' column is circled in red, highlighting the assignments for pins H17, J15, L16, M13, R15, and R17.

Name	Direction	Int...	Ne...	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive St
output	OUT			H17	<input checked="" type="checkbox"/>	15	LVCMOS33*	3.300		12
a	IN			J15	<input checked="" type="checkbox"/>	15	LVCMOS33*	3.300		
b	IN			L16	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
c	IN			M13	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
d	IN			R15	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
e	IN			R17	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		

Make sure you use LVCMOS33 as discussed in class as the I/O standard!

The screenshot shows the Vivado 2017.2.1 interface for a synthesized design. The I/O Ports table is highlighted, showing the following configuration:

Name	Direction	Int...	Ne...	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive St
output	OUT			H17	<input checked="" type="checkbox"/>	15	LVCMOS33*	3.300		12
a	IN			J15	<input checked="" type="checkbox"/>	15	LVCMOS33*	3.300		
b	IN			L16	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
c	IN			M13	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
d	IN			R15	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
e	IN			R17	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		

The I/O standard 'LVCMOS33*' is circled in red in the original image, indicating the correct configuration for the I/O ports.

When you save the design, you will be prompted for a constraints file name

The screenshot shows the Vivado 2017.2.1 interface. The 'File' menu is circled in red. The 'SYNTHESIZED DESIGN' window is active, showing a schematic diagram. A dialog box titled 'Out of Date Design' is displayed, with the following text:

Out of Date Design

Saving the current constraints to the target project constraints file may cause your synthesis to go out-of-date. To avoid re-running synthesis, you can force the design up-to-date by selecting the run in the Design Runs tab, right clicking, and selecting 'Force Up-to-Date'.

Don't show this dialog again

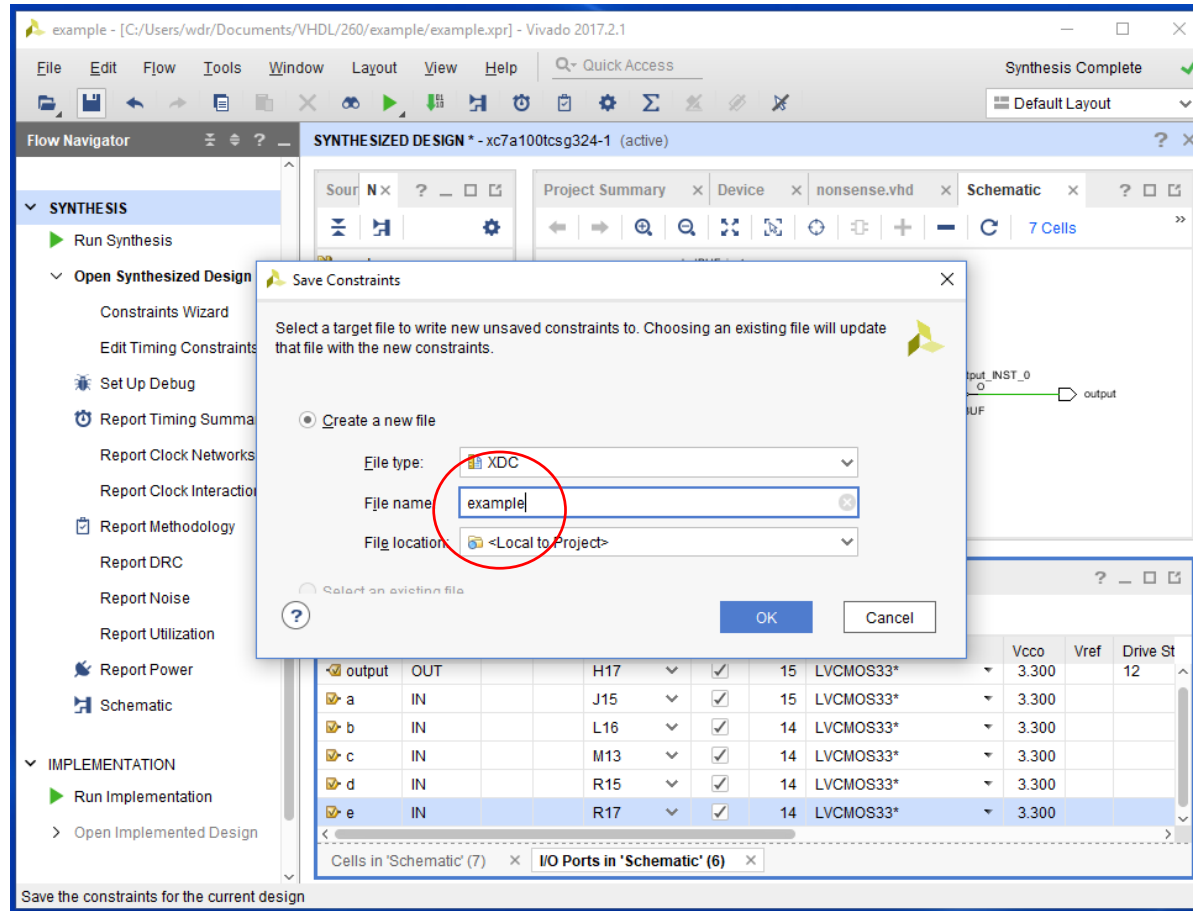
OK

Below the dialog box, the 'I/O Ports in Schematic' table is visible:

Name	Direction	Int...	Ne...	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive St
output	OUT			H17	<input checked="" type="checkbox"/>	15	LVC MOS33*	3.300		12
a	IN			J15	<input checked="" type="checkbox"/>	15	LVC MOS33*	3.300		
b	IN			L16	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300		
c	IN			M13	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300		
d	IN			R15	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300		
e	IN			R17	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300		

At the bottom of the window, the status bar reads: 'Save the constraints for the current design'.

When you save the design, you will be prompted for a constraints file name



The synthesis will now be out of date because of your pin assignments and I/O standard selection

The screenshot shows the Vivado 2017.2.1 interface. A red oval highlights the "Synthesis Out-of-date" warning in the top right corner. The main workspace displays a schematic diagram of a circuit with five input buffers (a-e) connected to a LUT5, which is connected to an output buffer (OBUF). The I/O Ports table at the bottom shows the following configuration:

Name	Direction	Int..	Ne..	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive St
output	OUT			H17	<input checked="" type="checkbox"/>	15	LVCMOS33*	3.300		12
a	IN			J15	<input checked="" type="checkbox"/>	15	LVCMOS33*	3.300		
b	IN			L16	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
c	IN			M13	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
d	IN			R15	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		
e	IN			R17	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300		

Run synthesis again...

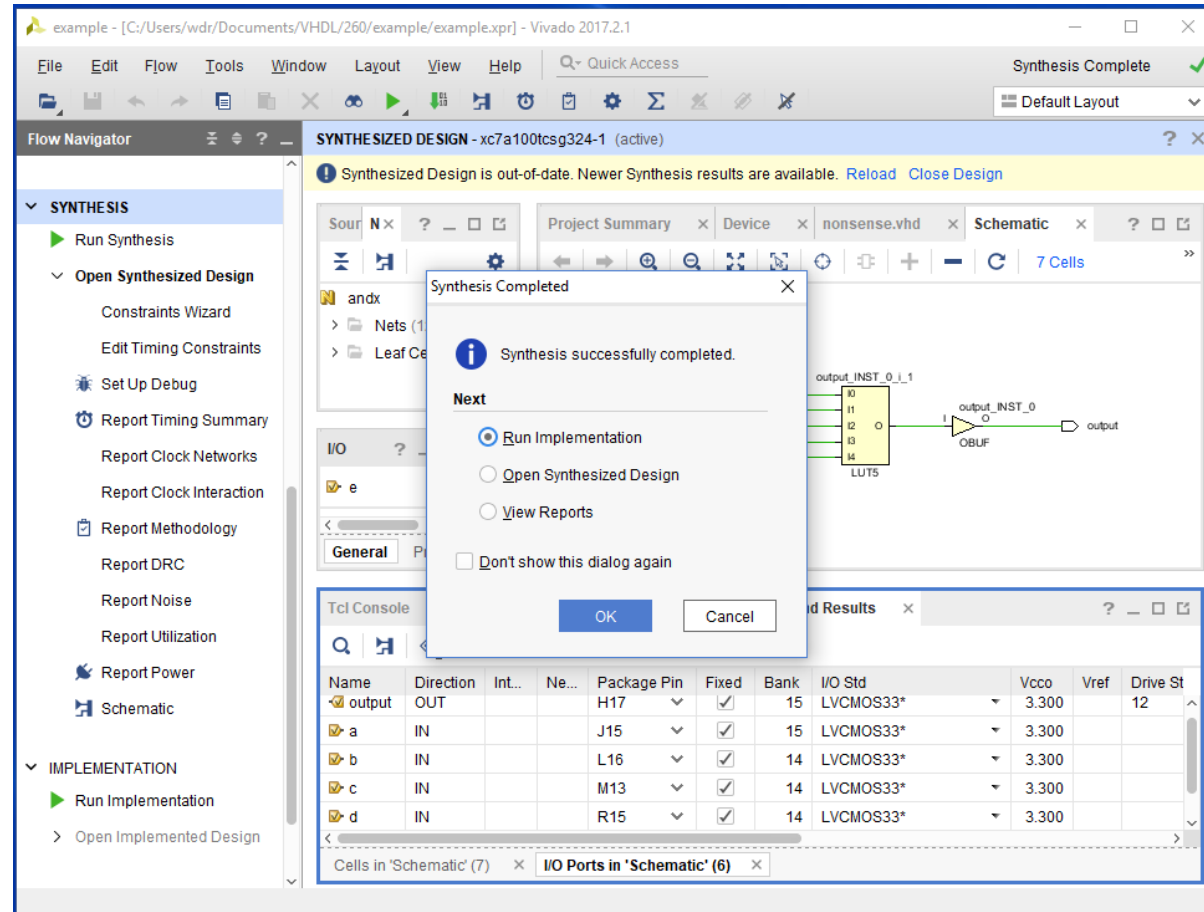
The screenshot shows the Vivado 2017.2.1 interface. The 'SYNTHESIS' menu is expanded, and the 'Run Synthesis' option is circled in red. A 'Launch Runs' dialog box is open, displaying the following options:

- Launch directory: <Default Launch Directory>
- Options:
 - Launch runs on local host: Number of jobs: 4
 - Generate scripts only
 - Don't show this dialog again

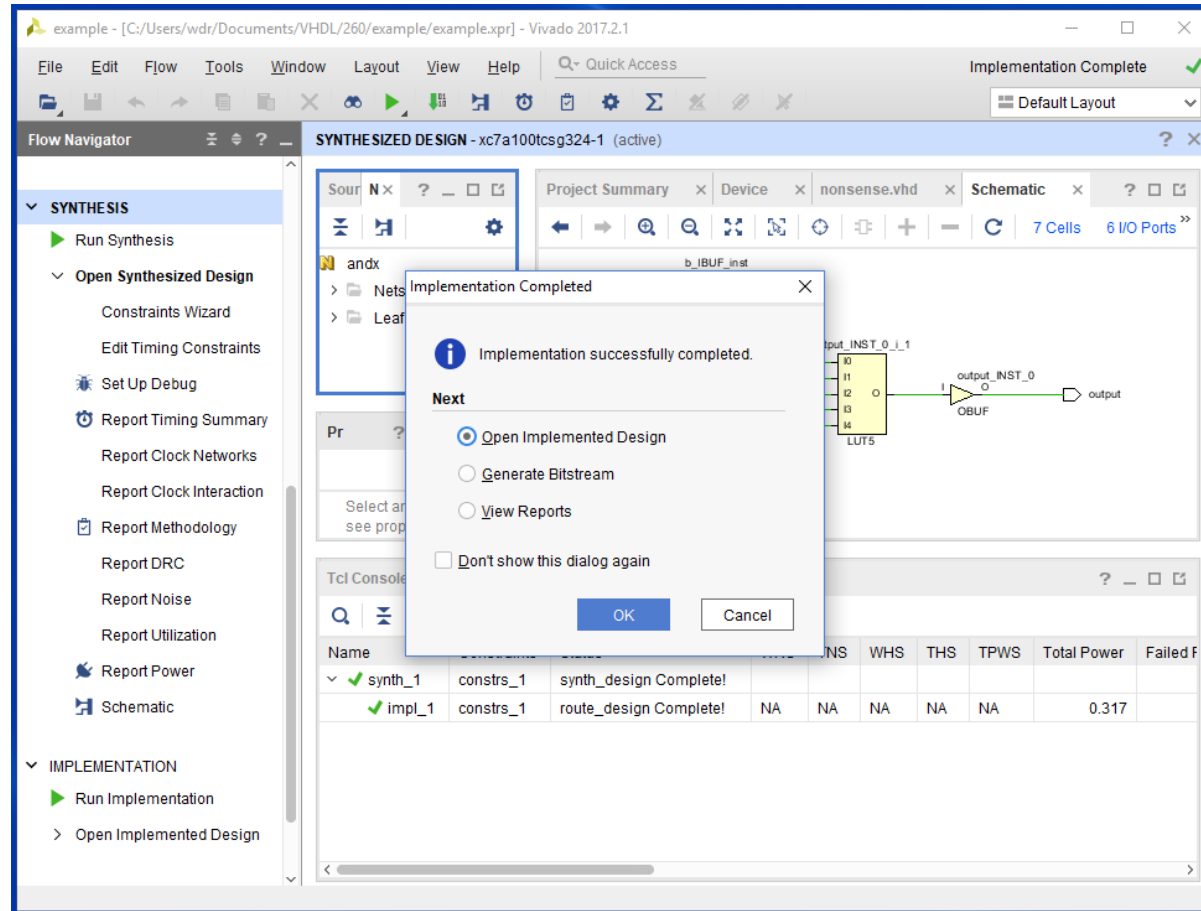
The dialog box has 'OK' and 'Cancel' buttons. In the background, a schematic diagram is visible, showing a signal path from '0_i_1' through an 'output_INST_0' block to an 'output' pin. Below the schematic, a table lists I/O ports:

Name	Direction	Pin	Package	Bank	Vcco	Vref	Drive St
output	OUT		H17	15	LVCMOS33*	3.300	12
a	IN	J15	15	LVCMOS33*	3.300		
b	IN	L16	14	LVCMOS33*	3.300		
c	IN	M13	14	LVCMOS33*	3.300		
d	IN	R15	14	LVCMOS33*	3.300		

This time, run select
“Run Implementation” by clicking “OK”



Select “Open Implemented Design” once place and route finishes



After looking at Reports, select “Generate Bitstream”

The screenshot shows the Vivado 2017.2.1 interface. The Flow Navigator on the left is expanded to the 'IMPLEMENTATION' section, where 'Generate Bitstream' is highlighted with a red circle. The main workspace displays the 'Netlist' view, the 'Project Summary' window with a utilization graph, and the 'Design Timing Summary' window.

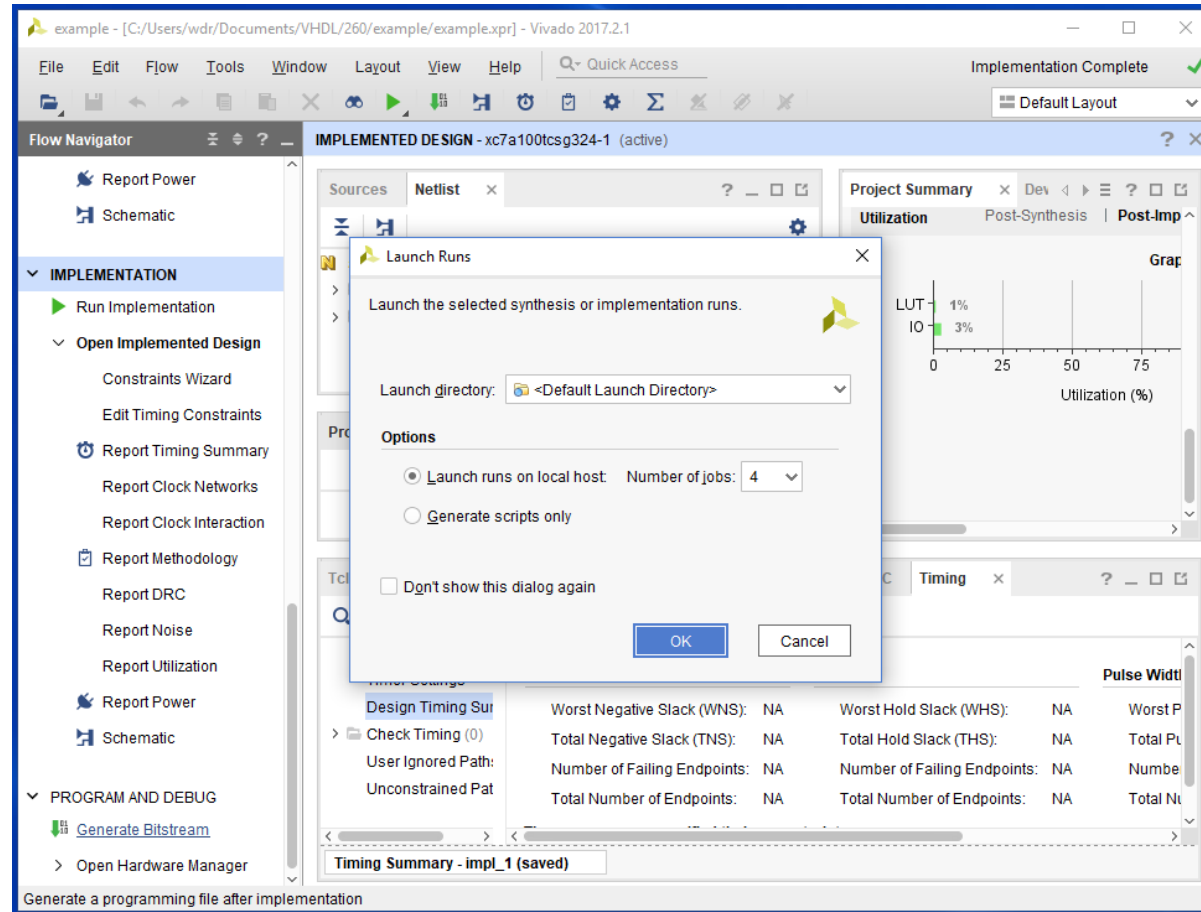
Project Summary - Utilization

Resource	Utilization (%)
LUT	1%
IO	3%

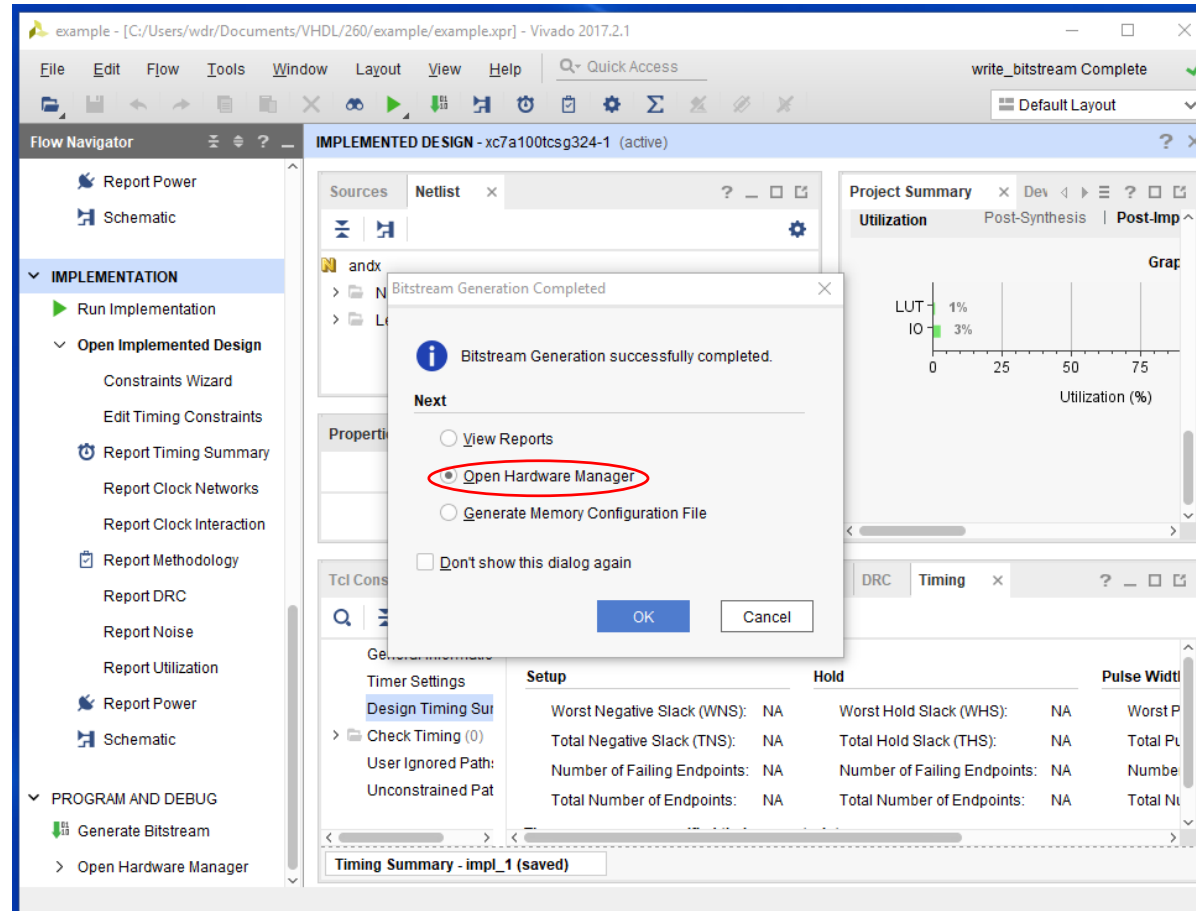
Design Timing Summary

General Information	Setup	Hold	Pulse Width
Worst Negative Slack (WNS):	NA	Worst Hold Slack (WHS):	NA
Total Negative Slack (TNS):	NA	Total Hold Slack (THS):	NA
Number of Failing Endpoints:	NA	Number of Failing Endpoints:	NA
Total Number of Endpoints:	NA	Total Number of Endpoints:	NA

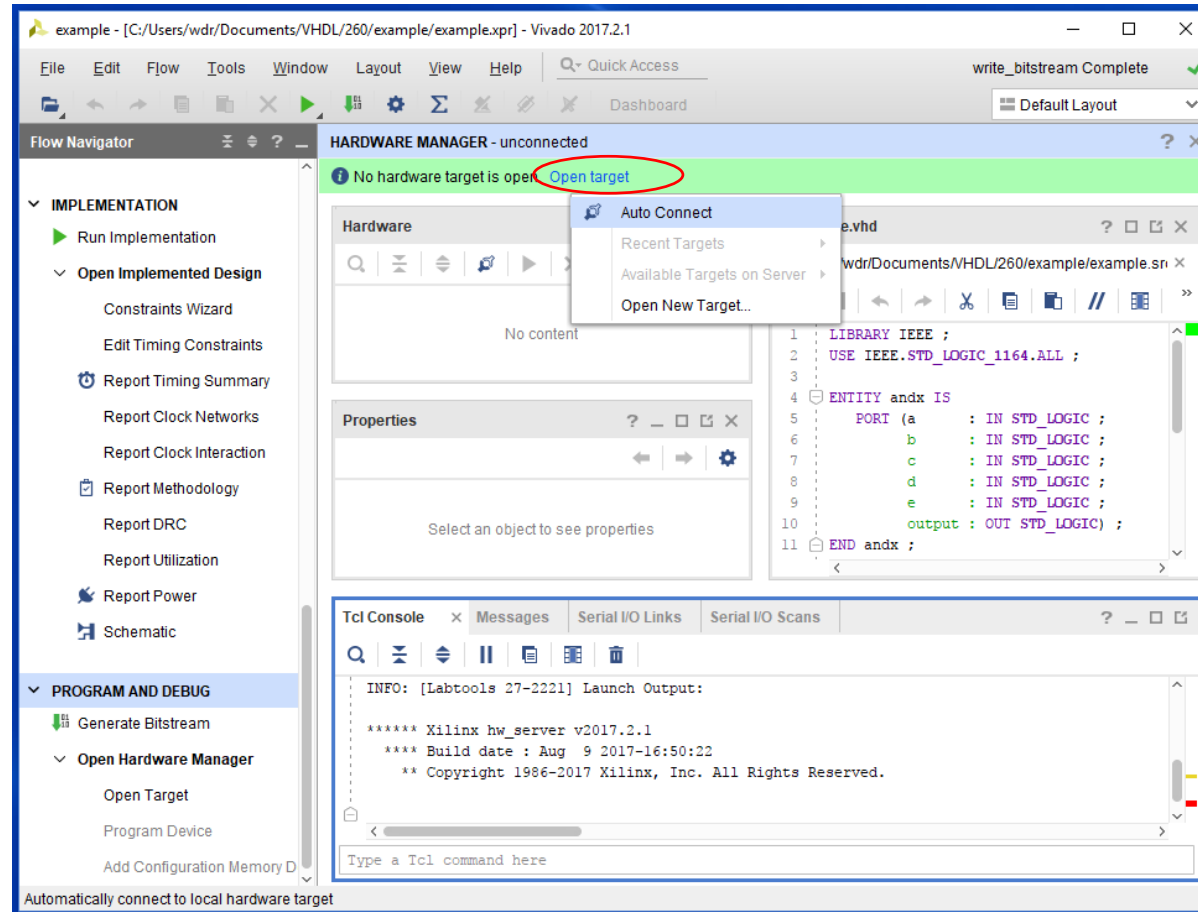
Clock “OK” at the info screen...



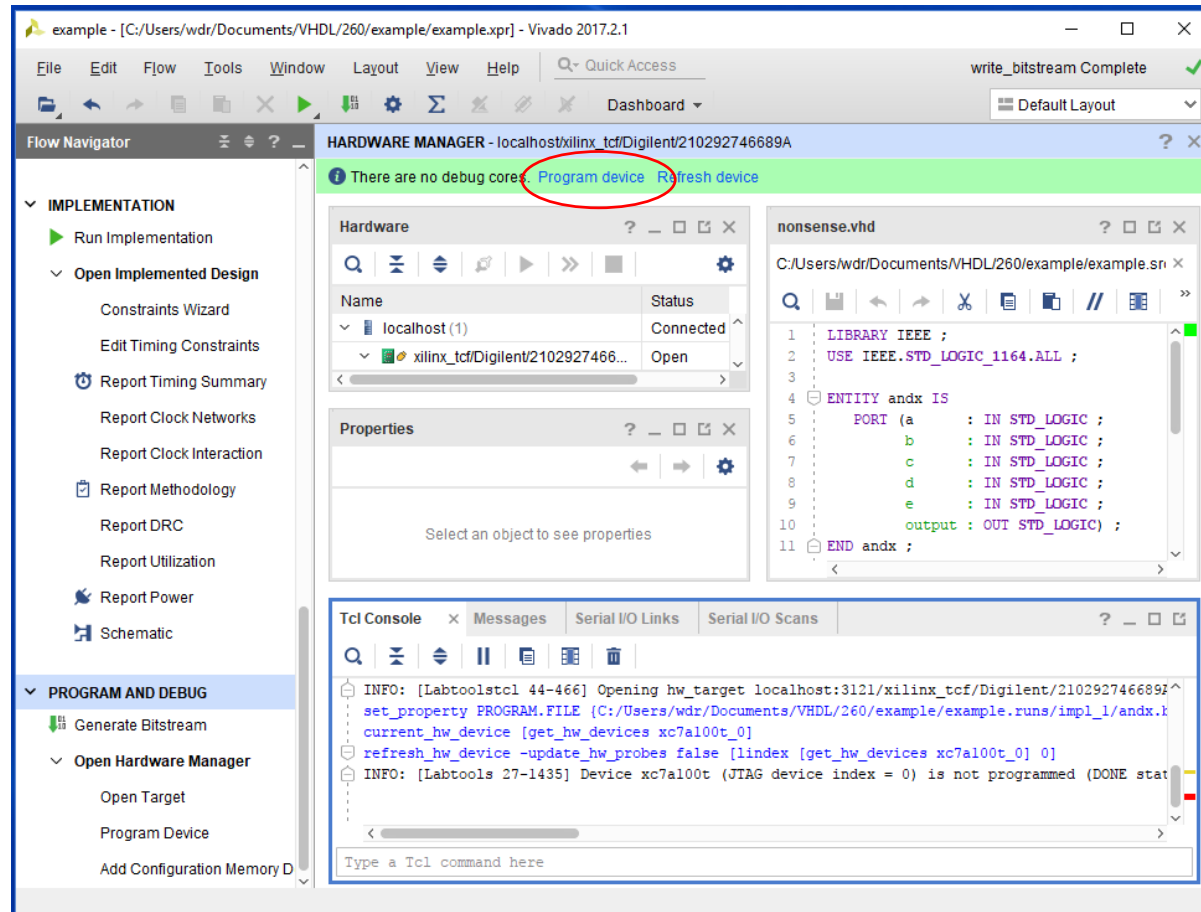
When Generate Bitstream finishes, select “Open Hardware Manager”



After plugging in the Digilent FPGA board to your laptop or PC, select “Open Target”



Select “Program device” after you see the Hardware Manager connect to the FPGA



Once the FPGA is programmed, you can test your design using the switches and LEDs!

