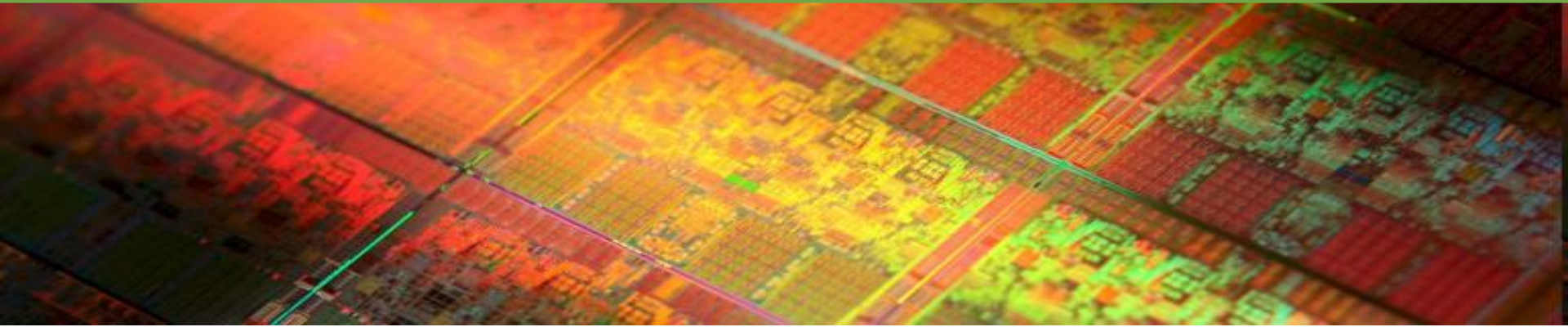


EE 2000

Logic Circuit Design (Fall 2018)



Tutorial 1

Xilinx Vivado Jumpstart

Instructor: Dr. Ray C.C. Cheung

Objectives

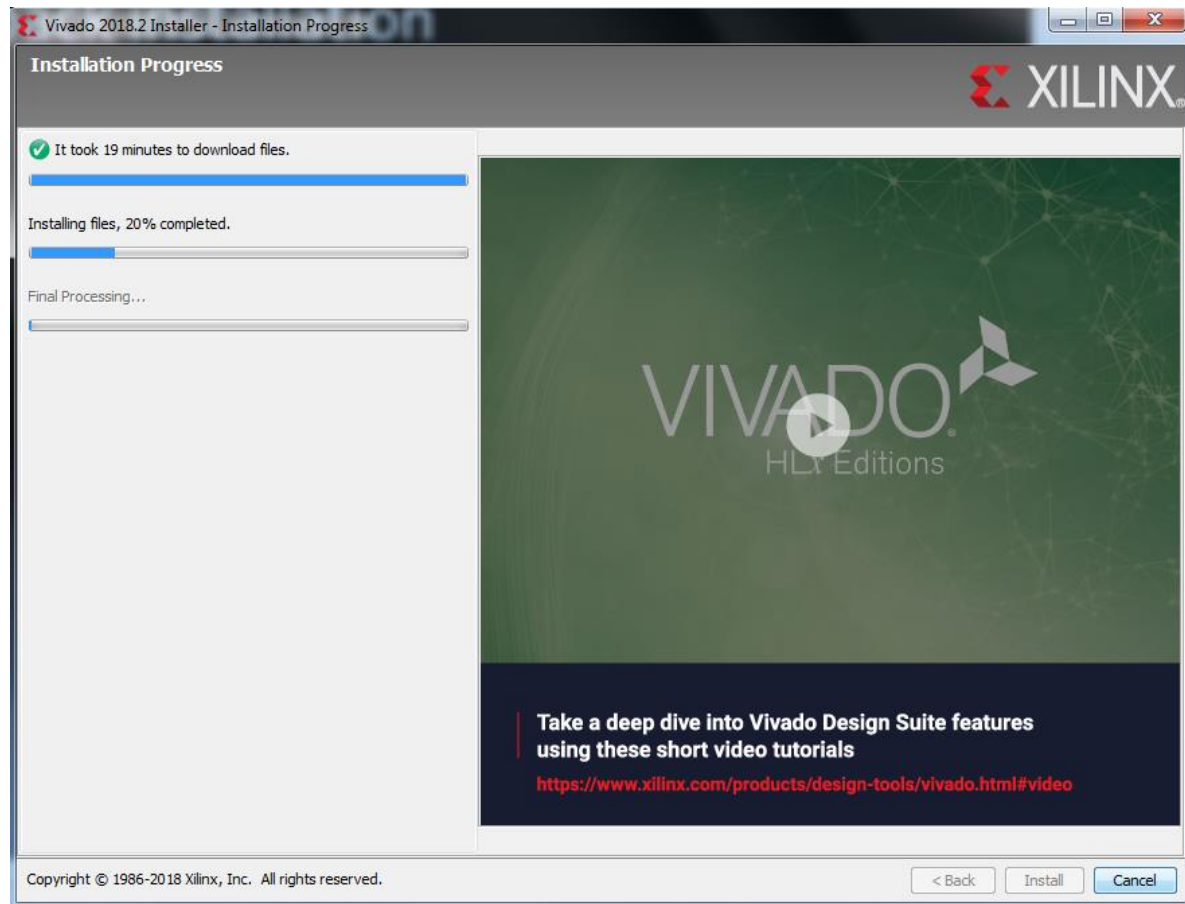
- Get a first feeling about this design tool
- Read the first VHDL code
 - A given example
- Experience different Vivado implementation steps
- Try to modify the VHDL Code

Download the software

- <https://www.xilinx.com/support/download.html>
- [Vivado HLx 2018.2: WebPACK and Editions - Windows Self Extracting Web Installer](#) (EXE - 50.56 MB)
- MD5 SUM Value : 1b00a58303ddb3bca5e84fa1b26685b0

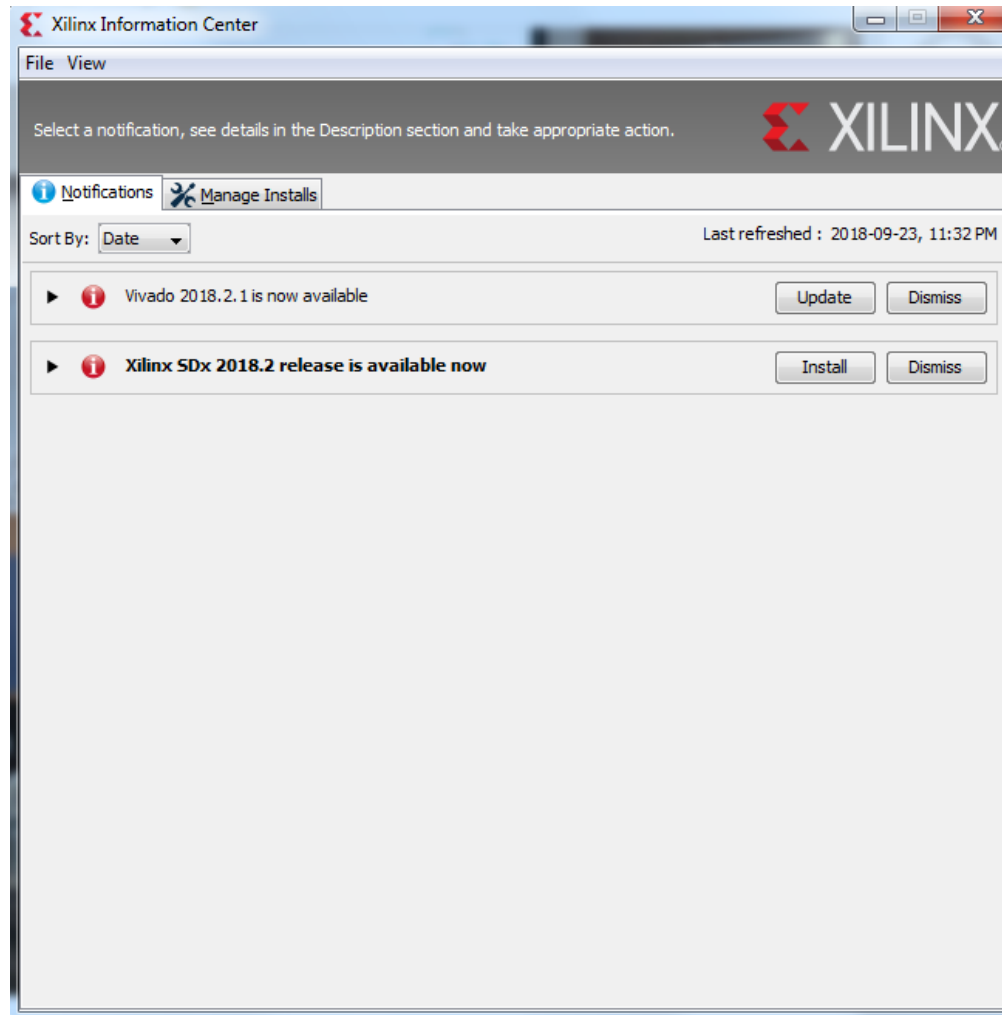
Install the software

- Select Webpack installation

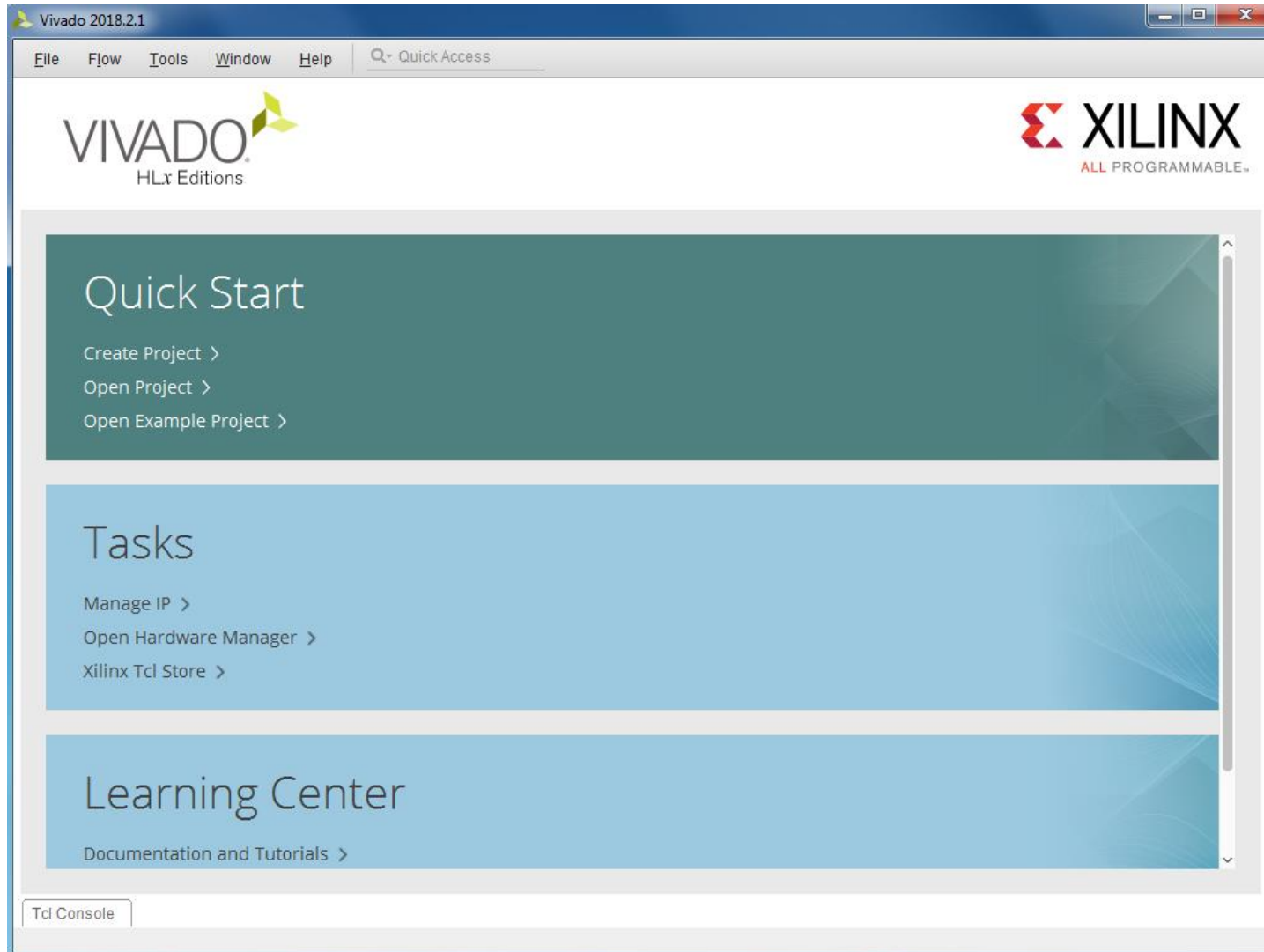


Update the software

- Check the update if necessary



Open a new Project



Step 1: Open the software

- Go to start-menu → Xilinx Vivado 2018
- Close any opened project
- Quick Start -> Create Project
- Project Name -> Tuto1
 - Select your location
- Select RTL Project
- Target Language -> VHDL
- Simulator Language -> VHDL

Project Manager - Overview

Tuto1 - [C:/Users/Dr RC/Tuto1/Tuto1.xpr] - Vivado 2018.2

File Edit Flow Tools Reports Window Layout View Help Q Quick Access

Flow Navigator PROJECT MANAGER - Tuto1

PROJECT MANAGER

- Settings
- Add Sources
- Language Templates
- IP Catalog

IP INTEGRATOR

- Create Block Design
- Open Block Design
- Generate Block Design

SIMULATION

- Run Simulation

RTL ANALYSIS

- Open Elaborated Design

SYNTHESIS

- Run Synthesis
- Open Synthesized Design

IMPLEMENTATION

- Run Implementation
- Open Implemented Design

PROGRAM AND DEBUG

- Generate Bitstream

Sources

Design Sources

Constraints

Simulation Sources

- sim_1

Hierarchy Libraries Compile Order

Properties

Select an object to see properties

Project Summary

Settings Edit

Project name: Tuto1

Project location: C:/Users/Dr RC/Tuto1

Product family: Kintex-7

Project part: xc7k70tffbv676-1

Top module name: Not defined

Target language: VHDL

Simulator language: VHDL

Synthesis

Status: Not started

Messages: No errors or warnings

Part: xc7k70tffbv676-1

Strategy: Vivado Synthesis Defaults

Report Strategy: Vivado Synthesis Default Reports

Implementation

Status: Not started

Messages: No errors or warnings

Part: xc7k70tffbv676-1

Strategy: Vivado Implementation Defaults

Report Strategy: Vivado Implementation Default Reports

Incremental compile: None

DRC Violations

[Run Implementation](#) to see DRC results

Timing

[Run Implementation](#) to see timing results

Utilization

[Run Synthesis](#) to see utilization results

Power

[Run Implementation](#) to see power results

Step 2: Add Source

- Select Design Sources -> Right click -> Add Sources
- Add design sources
- Create File -> VHDL
- Filename -> Tuto1 -> Click Finish
- Add ports
 - A -> Input
 - B -> Input
 - S -> Output

You can see this?

PROJECT MANAGER - Tuto1

Sources ? _ □ ▢ ✕

Q | ≡ | ≡ | + | ? | ● 0 ⚙

▼ Design Sources (1)

● **Tuto1**(Behavioral) (Tuto1.vhd)

> Constraints

▼ Simulation Sources (1)

> sim_1 (1)

Hierarchy

Libraries

Compile Order

Properties

? _ □ ▢ ✕

← → ⚙

Project Summary

Settings [Edit](#)

Project name: Tuto1

Project location: C:/Users/Dr RC/Tuto1

Product family: Kintex-7

Project part: [xc7k70tfbv676-1](#)

Top module name: [Tuto1](#)

Target language: [VHDL](#)

Simulator language: [VHDL](#)

Synthesis

Status: Not started

Messages: No errors or warnings

Part: xc7k70tfbv676-1

Strategy: [Vivado Synthesis Defaults](#)

Report Strategy: [Vivado Synthesis Default Reports](#)

Step 3: Create your AND Gate

- Double Click: Tuto1.VHD
- Please check if you see this “entity”
- entity Tuto1 is
 - Port (A : in STD_LOGIC;
 - B : in STD_LOGIC;
 - S : in STD_LOGIC);
- end Tuto1;

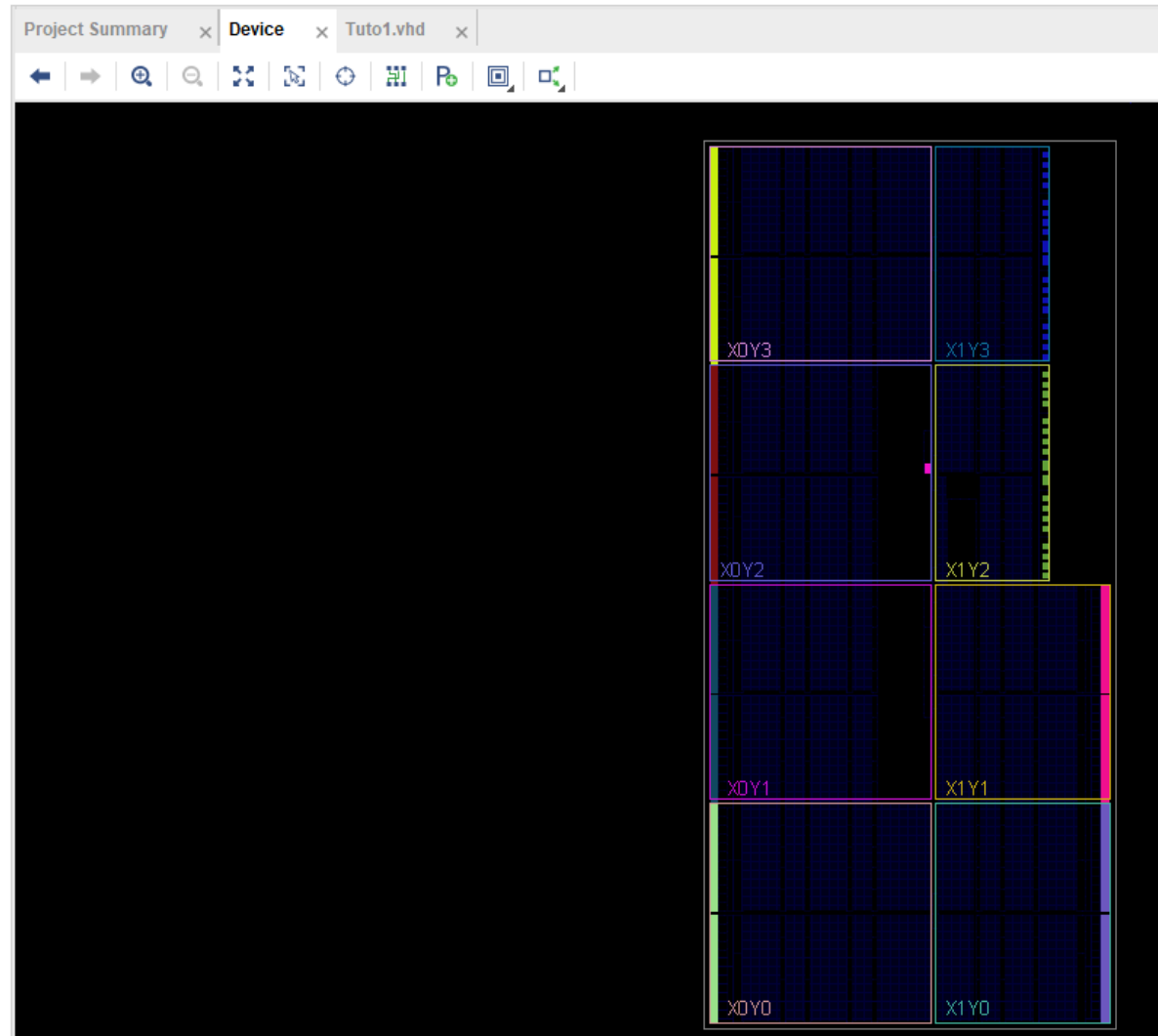
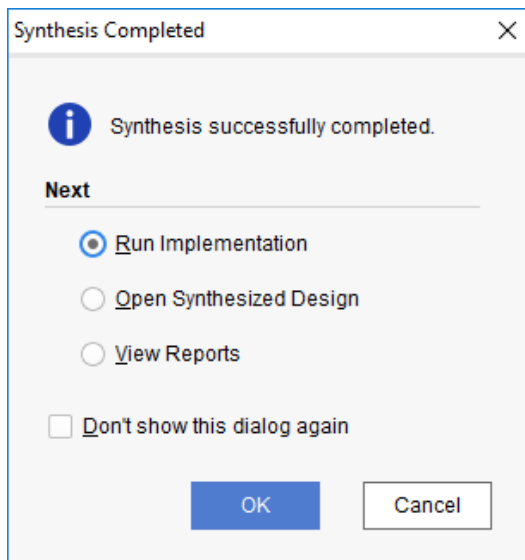
VHDL code - Simple AND Gate

- architecture Behavioral of Tuto1 is
- begin
- // Add your code below
- **S <= A and B;**
- end Behavioral;

Reading your first VHDL file

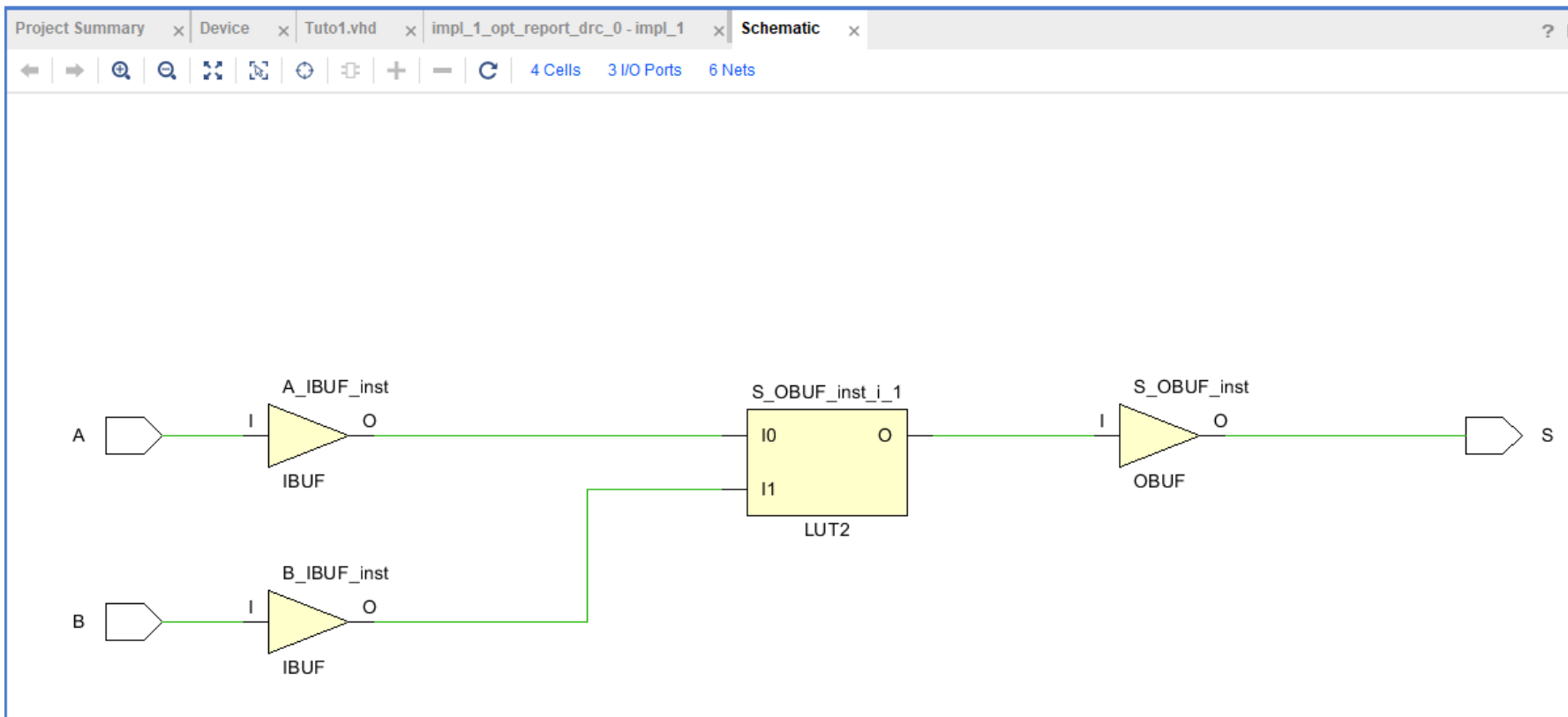
- How many lines of code are there?
- What is the first statement?
- What is the last statement?
- What else do you see?
- Go to “Synthesis” panel
 - Right click “Run Synthesis”

Check Synthesis Result



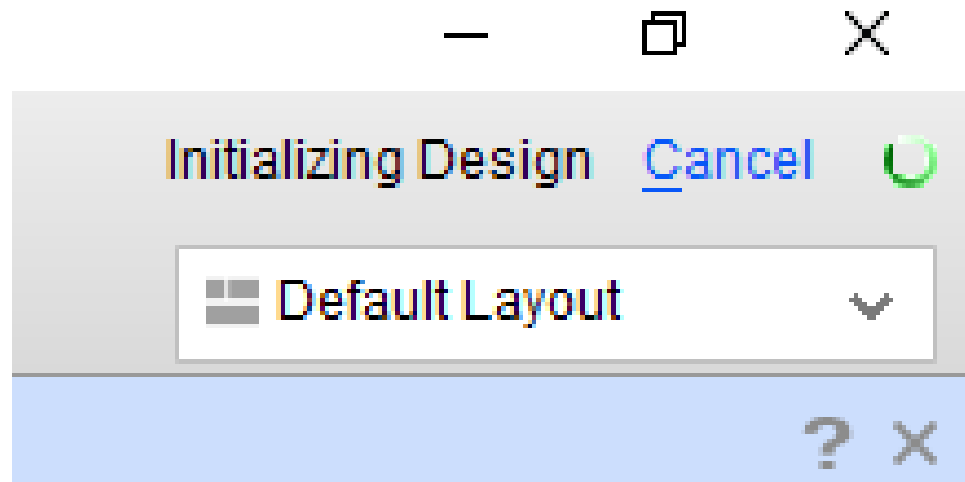
Check Schematic

- Is this what you want?



Check if your Program is running

- Rolling on the top right corner



Step 4: Run Implementation

▼ IMPLEMENTATION

▶ Run Implementation

▼ Open Implemented Design

Constraints Wizard

Edit Timing Constraints

🕒 Report Timing Summary

Report Clock Networks

Report Clock Interaction

📋 Report Methodology

Report DRC

Report Noise

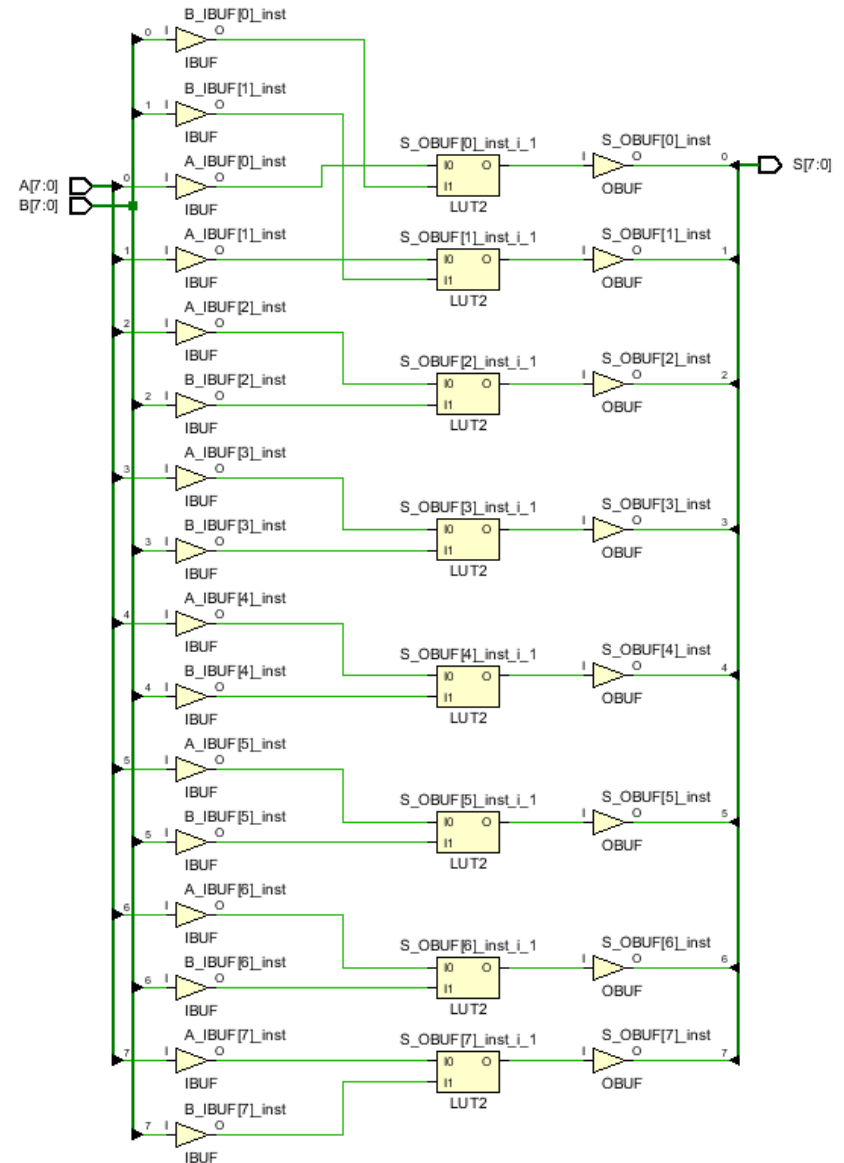
Report Utilization

Step 5: Change the Code from 1-bit to 8-bit

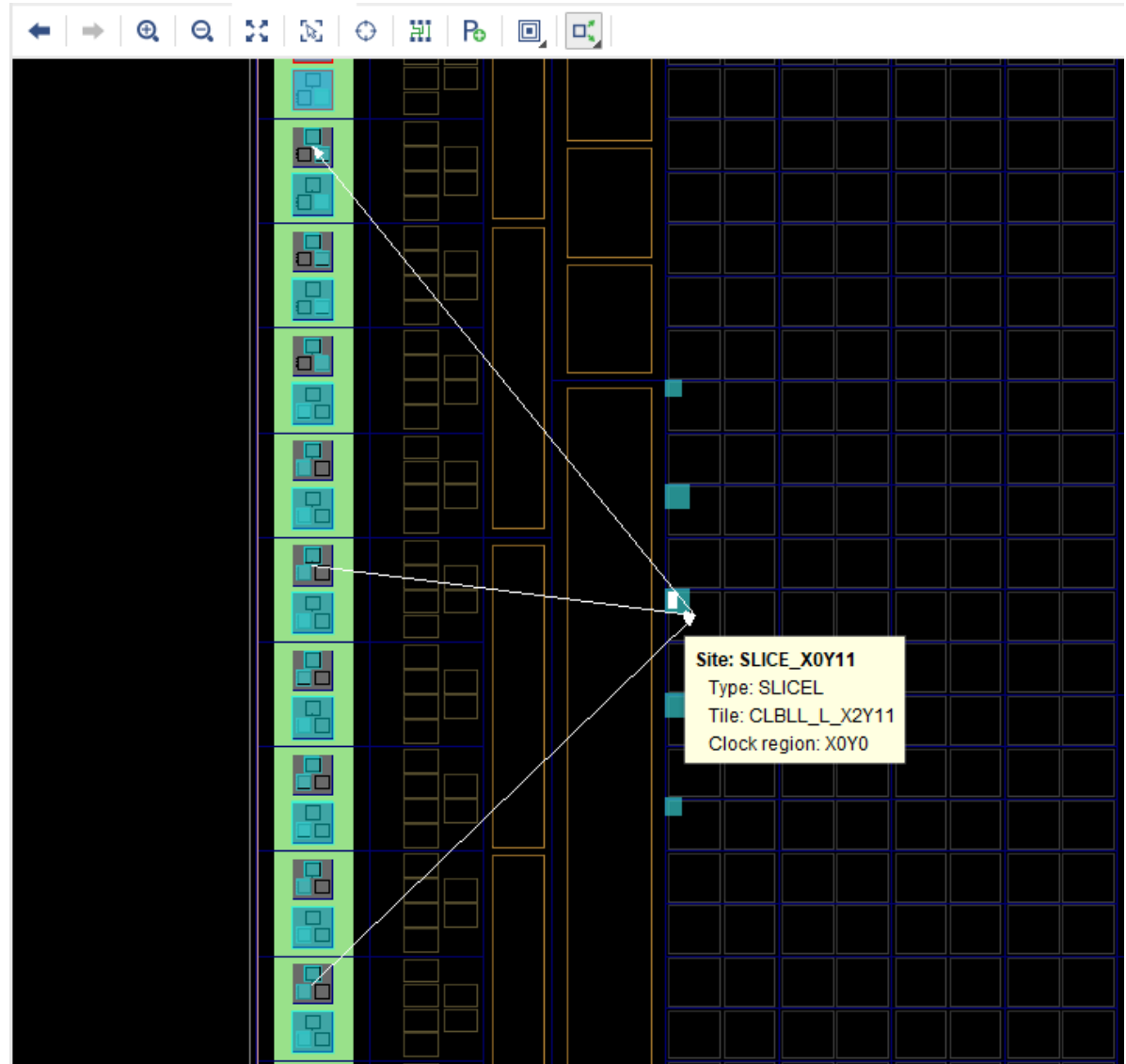
- entity Tuto1 is
- Port (A : in STD_LOGIC_VECTOR(7 downto 0);
- B : in STD_LOGIC_VECTOR(7 downto 0);
- S : out STD_LOGIC_VECTOR(7 downto 0));
- end Tuto1;
- The other parts remain unchanged.

8-bit AND Gate

- Are you able to get this?



Check the Slice and IO Buffer



Additional Xilinx Video

- <https://www.xilinx.com/video/technology/25g-interconnect-kintex-virtex-and-zynq-mpsoc.html>

Tutorial - Remarks

- Please raise your hand if you have questions!
 - 9 course tutors + course Instructors
 - If you finish the tutorial quickly, please try different settings in the Vivado tool.
- You can also refer to the course Canvas homepage.
- Get familiar with the Design tool.
- The Testbench and simulation will be covered in the next tutorial.