DIGITAL DESIGN

LAB1 USING VIVADO + FPGA DEVELOPMENT BOARD(MINISYS/EGO1)

2022 FALL TERM

TOPICS

- Experimental Platform

  EDA tool (Vivado 2017.4) + FPGA Development Board(EGO1/Minisys)

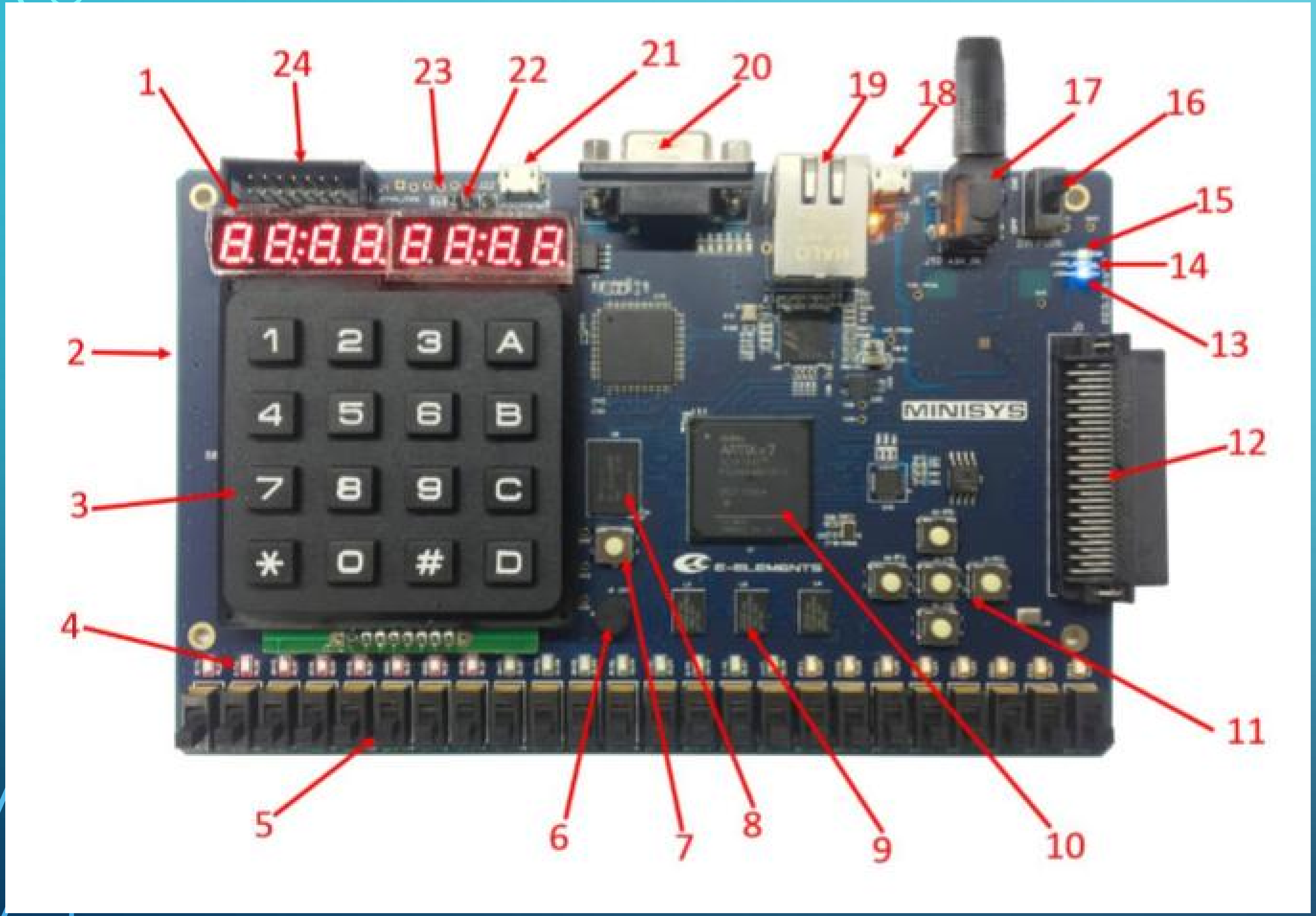
  - Vivado installation tips
  - FPGA Development Board(EGO1/Minisys) introduction
- 1st Lab on Digital Logic course
  - Build a Vivado project, add circuit design file and constraint file
  - Connect Vivado with FPGA Development Board and program the FPGA chip
  - Test the circuit which runs on the FPGA chip
- Questions and Exercises

EXPERIMENTAL PLATFORM: EDA + FPGA DEVELOPMENT BOARD

- Vivado (a type of EDA tools)：
  - Vivado is a design environment for FPGA products from Xilinx, and is tightly-coupled to the architecture of such chips, and cannot be used with FPGA products from other vendors.
  - Vivado enables developers to synthesize (compile) their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer.
  - The version we choose is Vivado 2017.4

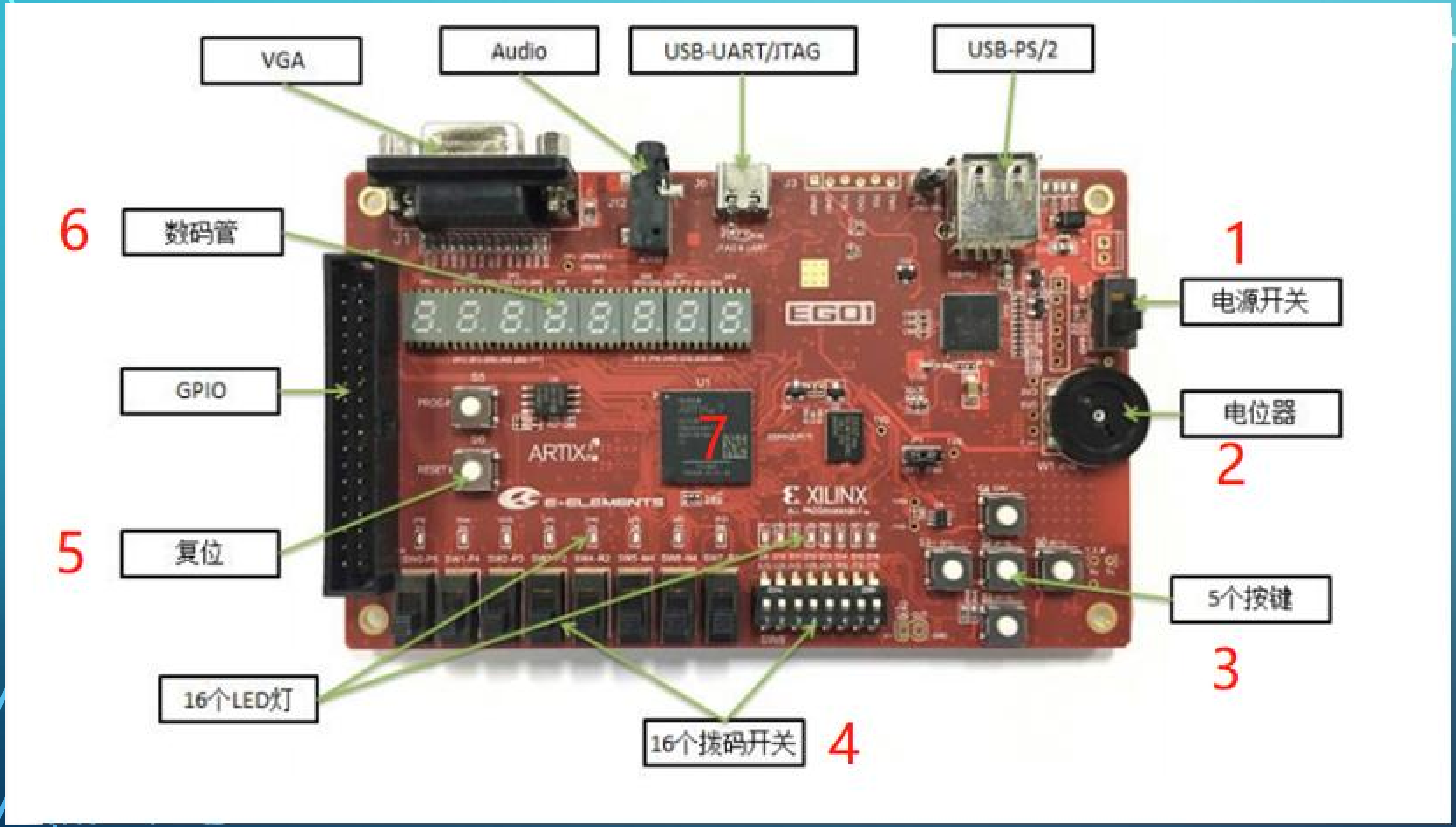
- Installation of Vivado (20 G free hard disk space is suggested)
  - Attention: the name of the directory which includes installation package MUST NOT containing Chinese and space characters.

FPGA DEVELOPMENT BOARD(MINISYS)



- Power Interface (17)
- USB-JTAG Interface (21)
- **Artix 7 FPGA chip** (10)
- SRAM (9)
- **Seven-Segment Digital Tube** (1)
- **Mini Keyboard** (3)
- **Dial Switch** *24 (5)
- **LED** *24 (4)

FPGA DEVELOPMENT BOARD(EGO1)



- Power Switch (1)
- Potentiometer (2)
- Button * 5 (3)
- Dial switch * 16 (4)
- **Reset Button (5)**
- **Seven-Segment Digital Tube * 8** (6)
- **Artix 7 FPGA chip** (7)

VIVADO(2017.4) INSTALLATION (TIPS1)

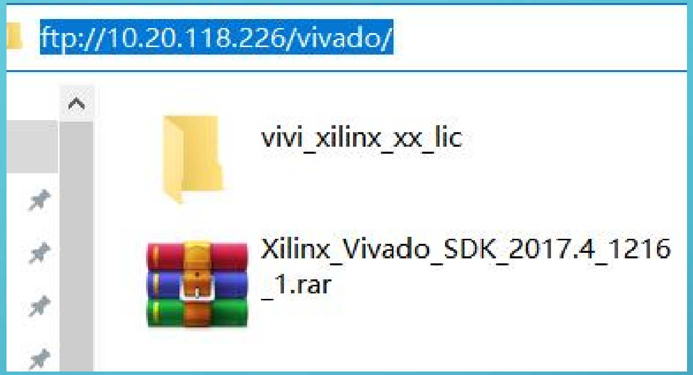
ftp://10.20.118.226/   account: ftp-d-logic   password: ggsddu

Download two files from FTP site to your PC:

Attention: Before coping, make sure there is enough free space in the destination disk.

1) Installation package: “Xinlinx_Vivado_SDK_2017.4_1216_1.rar”

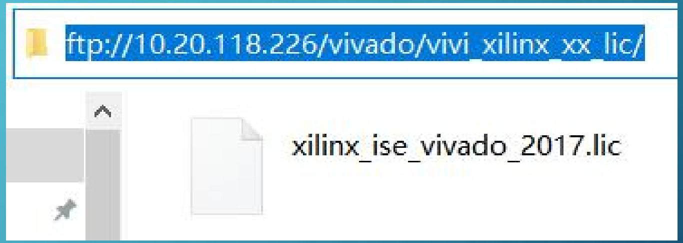
2) Lisence file: “xinlin_ise_vivado_2017.lic”



ftp://10.20.118.226/vivado/

vivi_xilinx_xx_lic

Xilinx_Vivado_SDK_2017.4_1216_1.rar



ftp://10.20.118.226/vivado/vivi_xilinx_xx_lic/

xilinx_ise_vivado_2017.lic

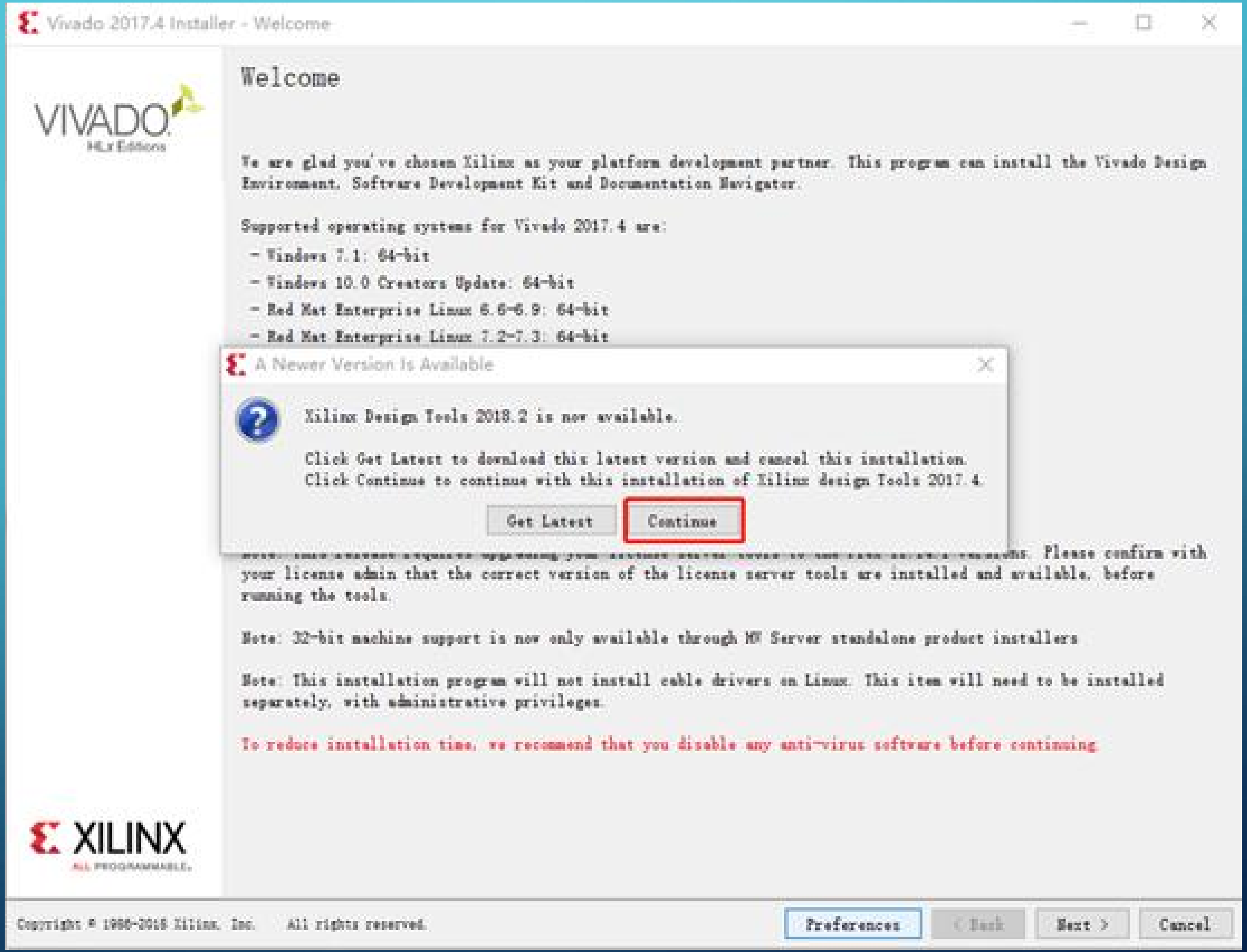
Decompress the compressed package“Xinlinx_Vivado_SDK_2017.4_1216_1.rar”, find the file “xsetup.exe”, **double click** it to start the installation.

xsetup

ATTENTION:
The FTP site is only valid for the user who are in Sustech compus.
If you are not in the compus, get all the files used for Vivado installation from the Baidu disk (webpage : https://pan.baidu.com/s/1MfeMCk2igcsf1WEQA49ObA   key: fhr4)

VIVADO(2017.4) INSTALLATION (TIPS2)

Following the steps below



Welcome

We are glad you've chosen Xilinx as your platform development partner. This program can install the Vivado Design Environment, Software Development Kit and Documentation Navigator.

Supported operating systems for Vivado 2017.4 are:

- Windows 7.1: 64-bit
- Windows 10.0 Creators Update: 64-bit
- Red Hat Enterprise Linux 6.6-6.9: 64-bit
- Red Hat Enterprise Linux 7.2-7.3: 64-bit

Xilinx Design Tools 2018.2 is now available.

Click Get Latest to download this latest version and cancel this installation.
Click Continue to continue with this installation of Xilinx design Tools 2017.4.

[illegible] Please confirm with your license admin that the correct version of the license server tools are installed and available, before running the tools.

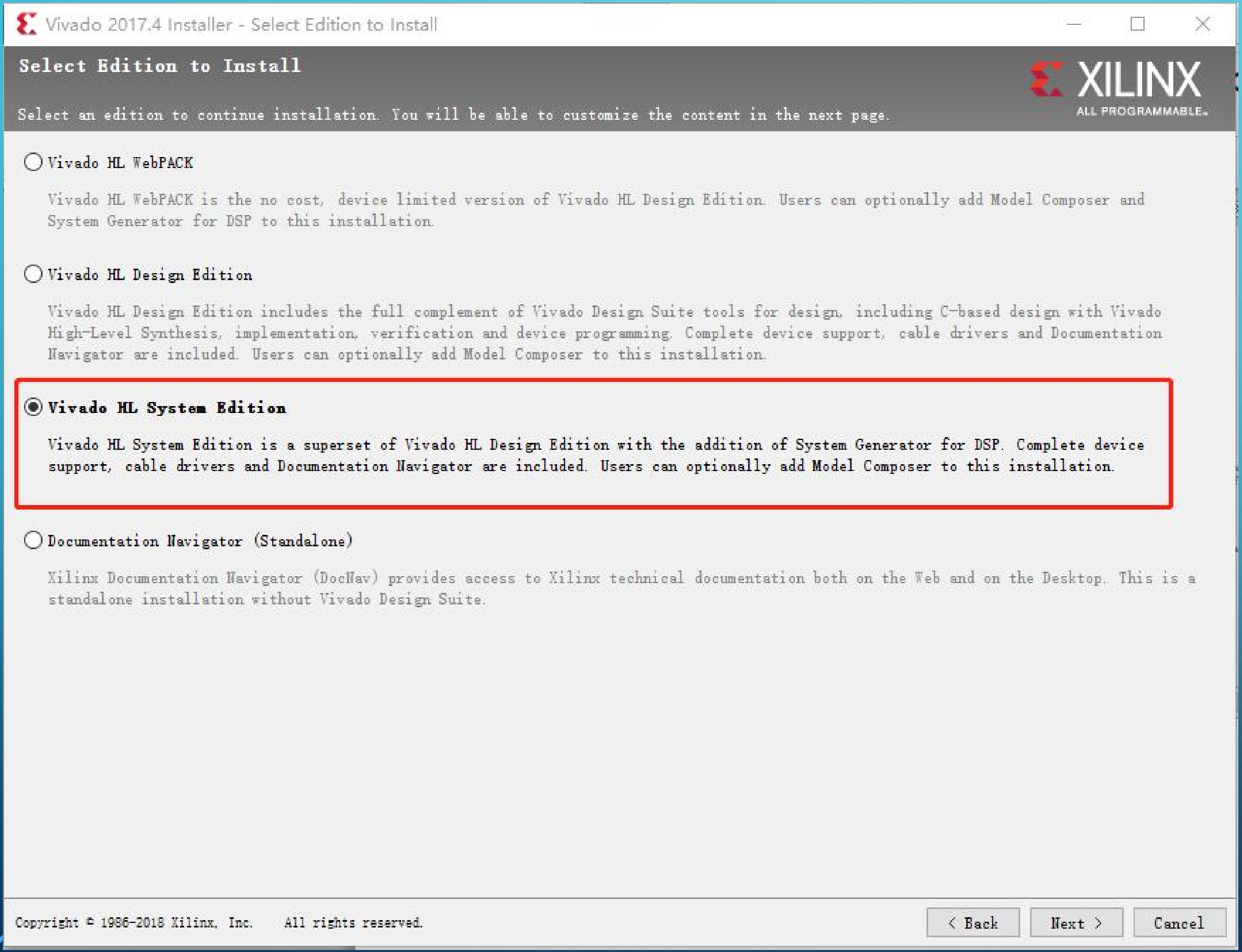
Note: 32-bit machine support is now only available through MW Server standalone product installers

Note: This installation program will not install cable drivers on Linux. This item will need to be installed separately, with administrative privileges.

To reduce installation time, we recommend that you disable any anti-virus software before continuing

Copyright © 1986-2018 Xilinx, Inc.   All rights reserved.

VIVADO(2017.4) INSTALLATION (TIPS3)



Select Edition to Install

Select an edition to continue installation. You will be able to customize the content in the next page.

○ Vivado HL WebPACK

Vivado HL WebPACK is the no cost, device limited version of Vivado HL Design Edition. Users can optionally add Model Composer and System Generator for DSP to this installation.

○ Vivado HL Design Edition

Vivado HL Design Edition includes the full complement of Vivado Design Suite tools for design, including C-based design with Vivado High-Level Synthesis, implementation, verification and device programming. Complete device support, cable drivers and Documentation Navigator are included. Users can optionally add Model Composer to this installation.

● **Vivado HL System Edition**

Vivado HL System Edition is a superset of Vivado HL Design Edition with the addition of System Generator for DSP. Complete device support, cable drivers and Documentation Navigator are included. Users can optionally add Model Composer to this installation.

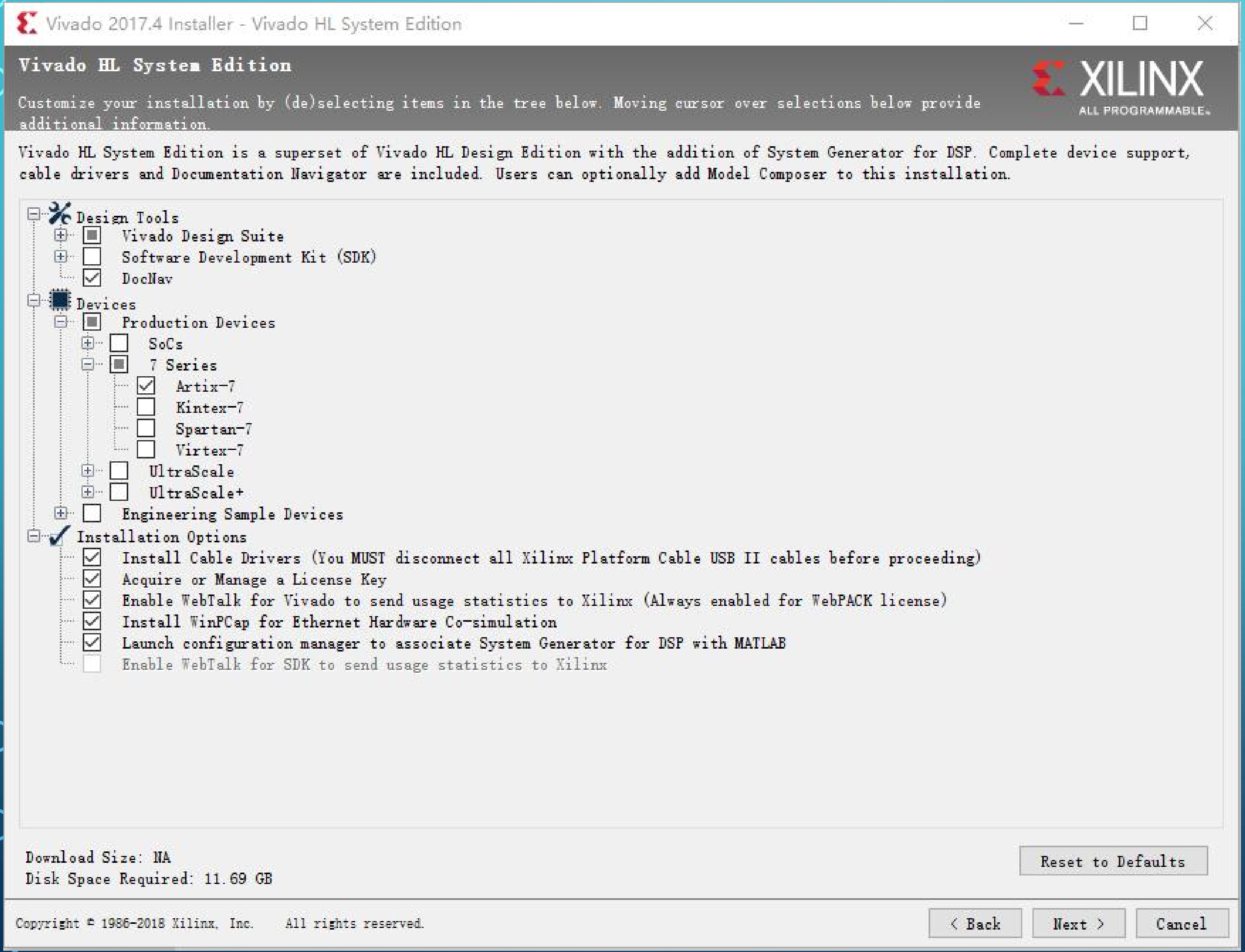
○ Documentation Navigator (Standalone)

Xilinx Documentation Navigator (DocNav) provides access to Xilinx technical documentation both on the Web and on the Desktop. This is a standalone installation without Vivado Design Suite.

Copyright © 1986-2018 Xilinx, Inc. All rights reserved.

It’s suggested to install the ”Vivado HL System Edition” because of its “complete device support” and “Documention Navigator” .

VIVADO(2017.4) INSTALLING (TIPS4)



Vivado HL System Edition

Customize your installation by (de)selecting items in the tree below. Moving cursor over selections below provide additional information.

Vivado HL System Edition is a superset of Vivado HL Design Edition with the addition of System Generator for DSP. Complete device support, cable drivers and Documentation Navigator are included. Users can optionally add Model Composer to this installation.

- Design Tools
  - ■ Vivado Design Suite
  - □ Software Development Kit (SDK)
  - ☑ DocNav
- Devices
  - ■ Production Devices
    - □ SoCs
    - ■ 7 Series
      - ☑ Artix-7
      - □ Kintex-7
      - □ Spartan-7
      - □ Virtex-7
    - □ UltraScale
    - □ UltraScale+
  - □ Engineering Sample Devices
- ✓ Installation Options
  - ☑ Install Cable Drivers (You MUST disconnect all Xilinx Platform Cable USB II cables before proceeding)
  - ☑ Acquire or Manage a License Key
  - ☑ Enable WebTalk for Vivado to send usage statistics to Xilinx (Always enabled for WebPACK license)
  - ☑ Install WinPCap for Ethernet Hardware Co-simulation
  - ☑ Launch configuration manager to associate System Generator for DSP with MATLAB
  - □ Enable WebTalk for SDK to send usage statistics to Xilinx

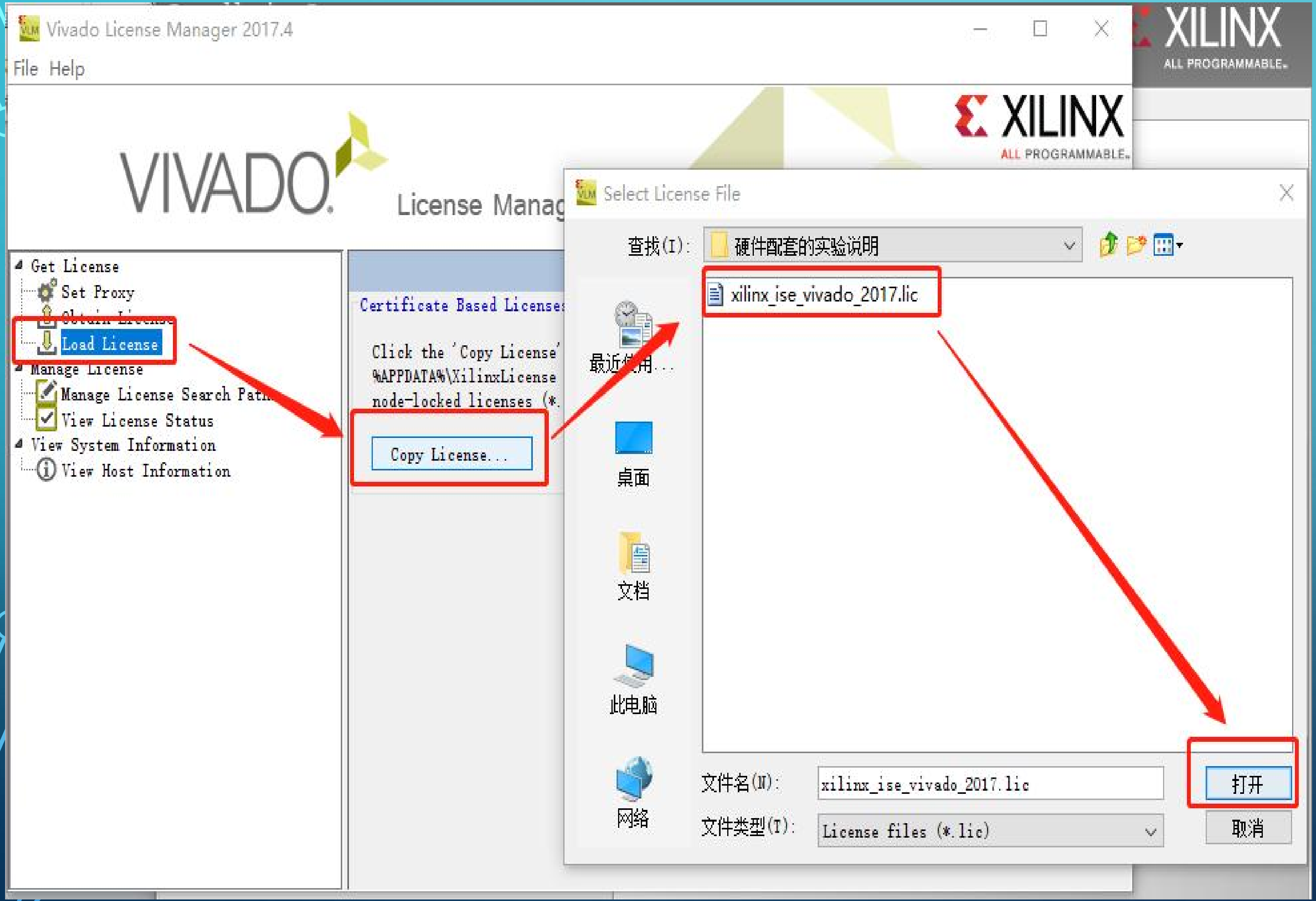
Download Size: NA
Disk Space Required: 11.69 GB

Copyright © 1986-2018 Xilinx, Inc. All rights reserved.

The size of “Vivado HL system Edition” is huge.
It is strongly recommended to select **ONLY WHAT IS NEEDED TO INSTALL**!!

The options in the left figure are enough for both the Digitial Logic course and Computer Orgnization course.

VIVADO INSTALLING (TIPS4)



Click the 'Copy License'
%APPDATA%\XilinxLicense
node-locked licenses (*.

文件名(N): xilinx_ise_vivado_2017.lic

文件类型(T): License files (*.lic)

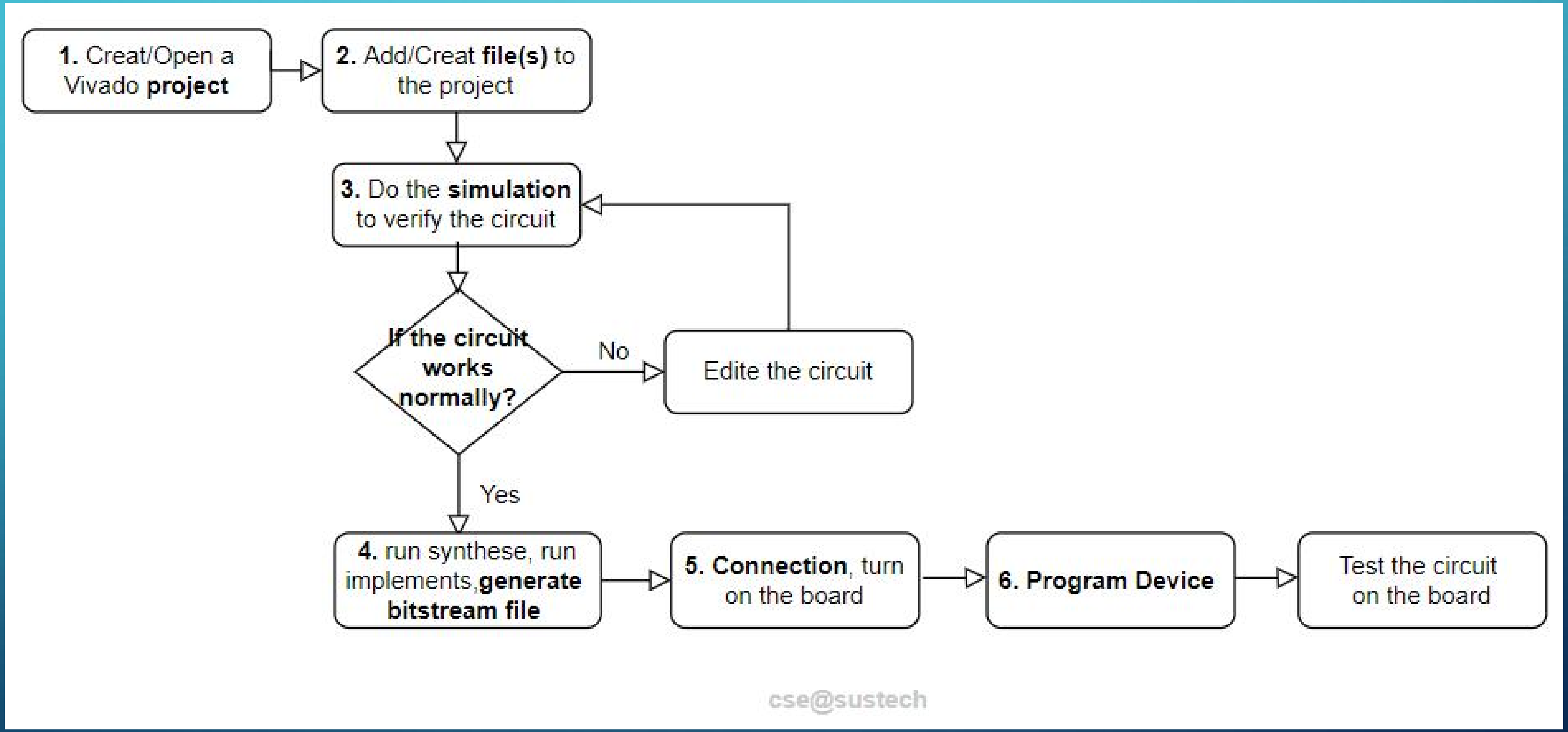
At the end of the installing, **load license** as shown in the left figure**.**

ATTENTION:
The license file(with ".lic" as its suffix) could be in **ANYWHERE** of your computer which is now doing the Vivado installation.

"硬件配置的实验说明" here is just a demo on my computer, **NOT MUST.**

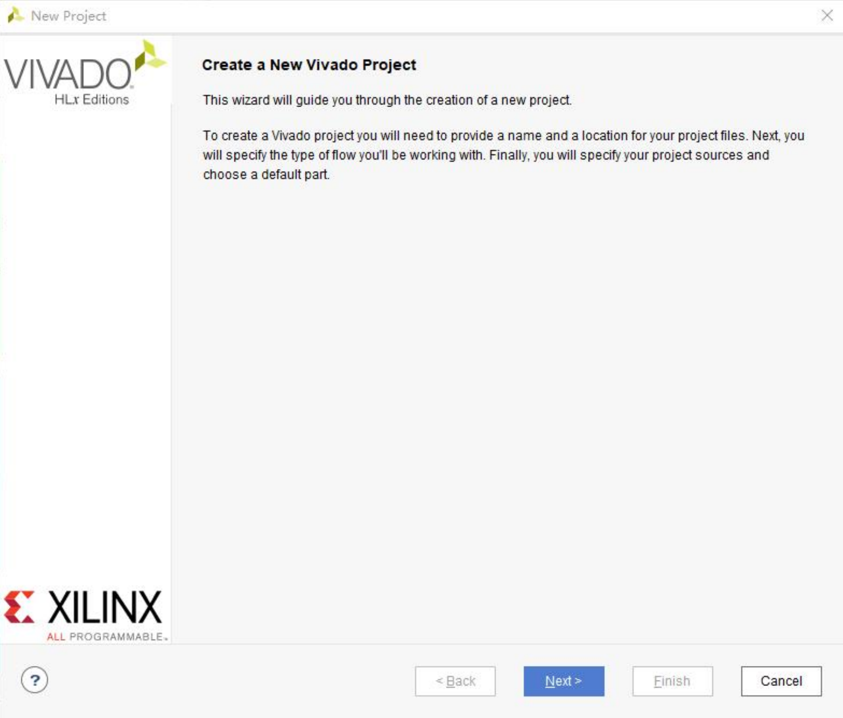
USING VIVADO + FPGA DEVELOPMENT BOARD

Follow the Following steps to make your designed circuit implement on the FPGA chip, test the circuit on the board which is embeded with the FPGA chip and other input/output device(s).


1. Creat/Open a Vivado **project**
2. Add/Creat **file(s)** to the project
3. Do the **simulation** to verify the circuit
If the circuit works normally?
No
Edite the circuit
Yes
4. run synthese, run implements,**generate bitstream file**
5. **Connection**, turn on the board
6. **Program Device**
Test the circuit on the board
cse@sustech

USING VIVADO + FPGA DEVELOPMENT BOARD(1)

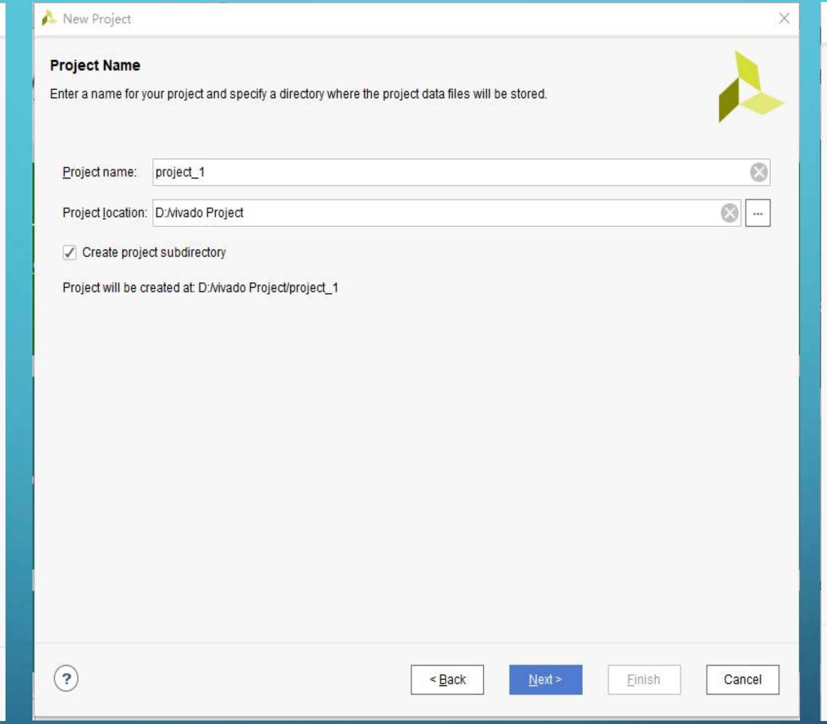
STEP1-1. **Create project,** select “rtl type” as its **type,** determin the **project’s name and location**(the name and location of the project MUST NOT containing Chinese characters), select the corresponding FPGA chip name.



Create a New Vivado Project

This wizard will guide you through the creation of a new project.

To create a Vivado project you will need to provide a name and a location for your project files. Next, you will specify the type of flow you'll be working with. Finally, you will specify your project sources and choose a default part.



Project Name

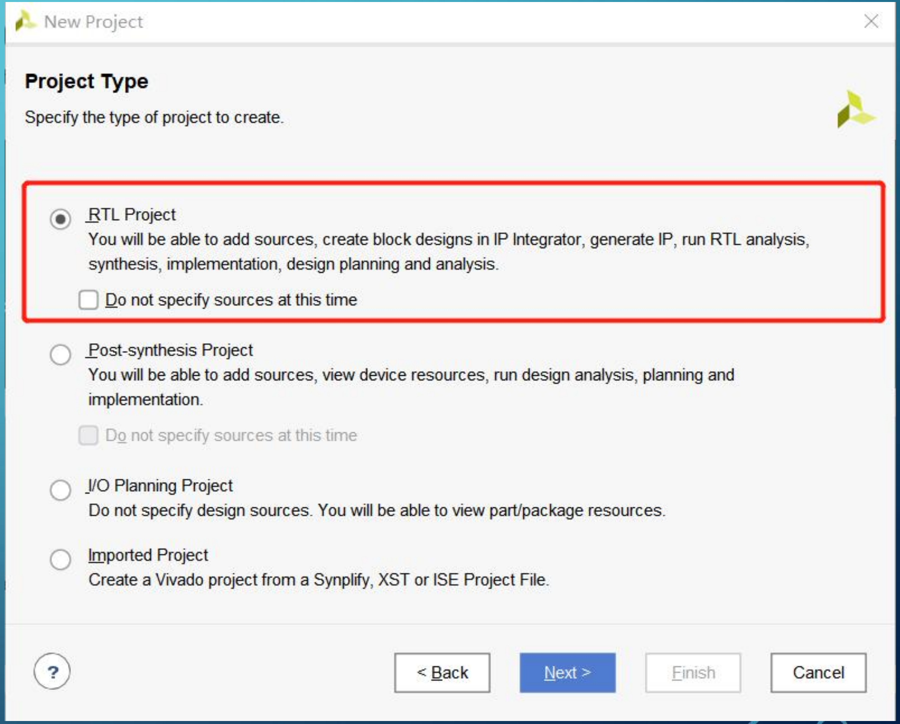
Enter a name for your project and specify a directory where the project data files will be stored.

Project name: project_1

Project location: D:/vivado Project

☑ Create project subdirectory

Project will be created at: D:/vivado Project/project_1



Project Type

Specify the type of project to create.

● RTL Project
You will be able to add sources, create block designs in IP Integrator, generate IP, run RTL analysis, synthesis, implementation, design planning and analysis.
□ Do not specify sources at this time

○ Post-synthesis Project
You will be able to add sources, view device resources, run design analysis, planning and implementation.
□ Do not specify sources at this time

○ I/O Planning Project
Do not specify design sources. You will be able to view part/package resources.

○ Imported Project
Create a Vivado project from a Synplify, XST or ISE Project File.

Attention:

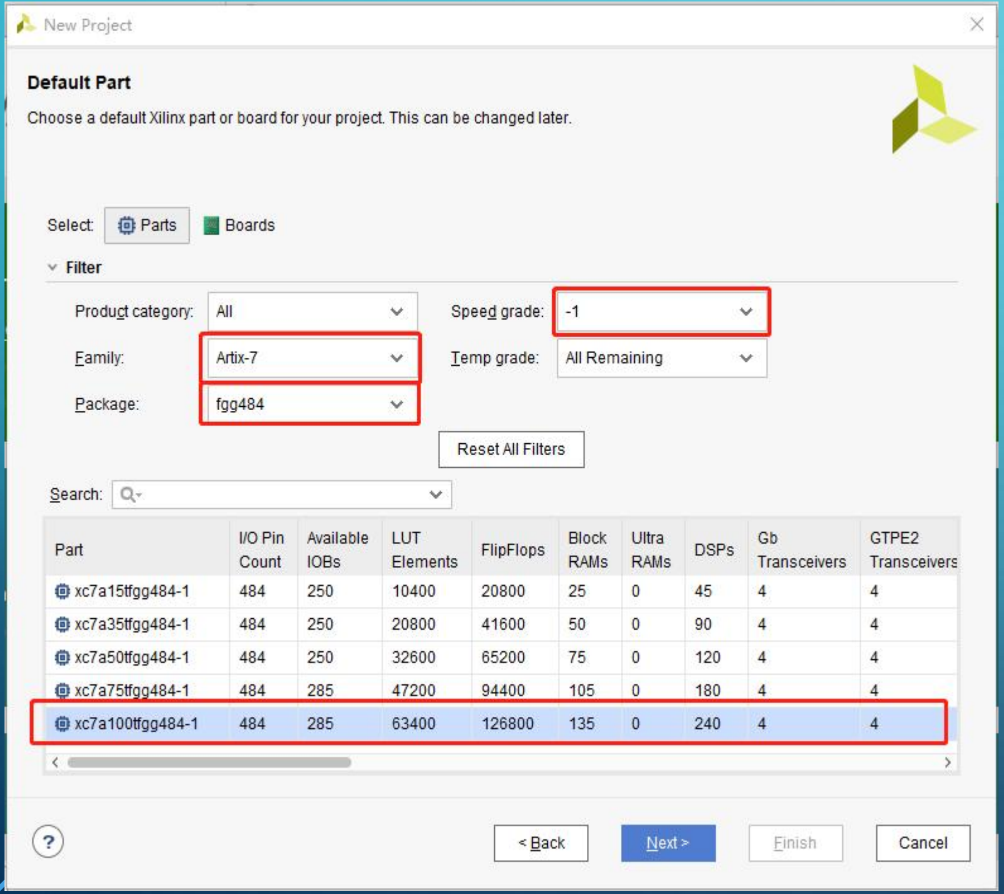
the value of project name and location in the middle figure is just a demo, all of them should be upto you to detcide.

USING VIVADO + FPGA DEVELOPMENT BOARD(2)

STEP1-2. select the corresponding FPGA chip. The version of FPGA chip in EGO1 is different from which in Minisys board.

The version of FPGA Chip in Minisys board:
Artix 7 xc7a100t fgg484-1

The version of FPGA Chip in EGO1 board:
Artix 7 xc7a35t CSG324-1



New Project

Default Part

Choose a default Xilinx part or board for your project. This can be changed later.

Select: Parts | Boards

Filter

Product category: All

Speed grade: -1

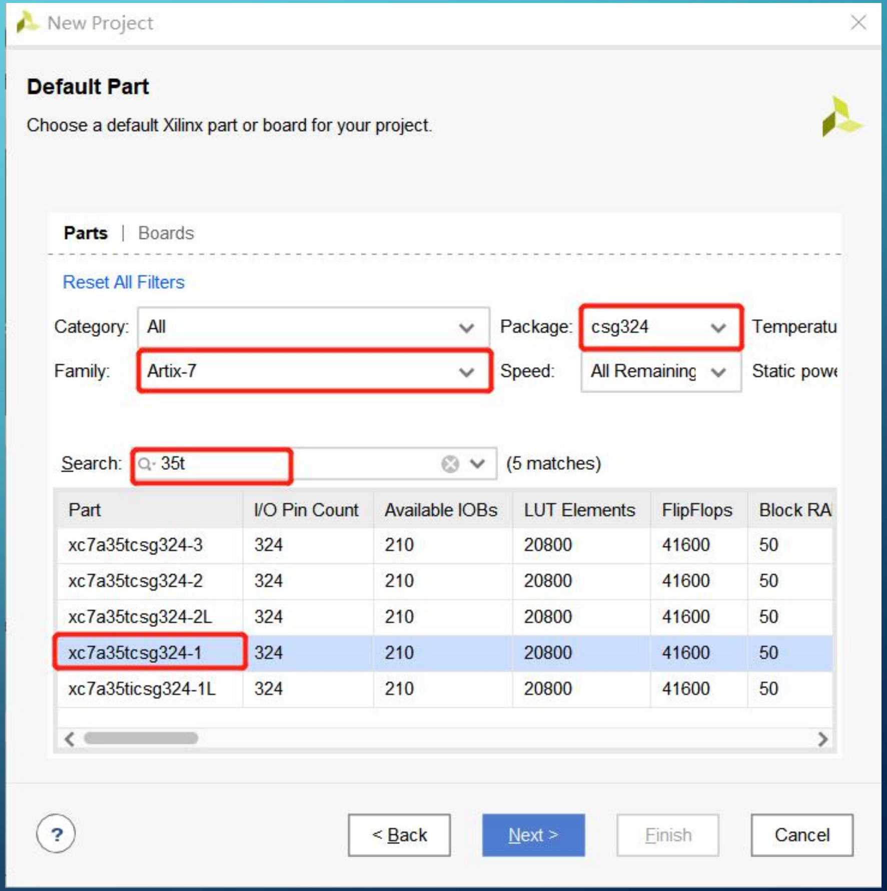
Family: Artix-7

Temp grade: All Remaining

Package: fgg484

Search:

Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	Ultra RAMs	DSPs	Gb Transceivers	GTPE2 Transceivers
xc7a15tfgg484-1	484	250	10400	20800	25	0	45	4	4
xc7a35tfgg484-1	484	250	20800	41600	50	0	90	4	4
xc7a50tfgg484-1	484	250	32600	65200	75	0	120	4	4
xc7a75tfgg484-1	484	285	47200	94400	105	0	180	4	4
xc7a100tfgg484-1	484	285	63400	126800	135	0	240	4	4



New Project

Default Part

Choose a default Xilinx part or board for your project.

Parts | Boards

Reset All Filters

Category: All

Package: csg324

Temperatu

Family: Artix-7

Speed: All Remaining

Static powe

Search: 35t (5 matches)

Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RA
xc7a35tcsg324-3	324	210	20800	41600	50
xc7a35tcsg324-2	324	210	20800	41600	50
xc7a35tcsg324-2L	324	210	20800	41600	50
xc7a35tcsg324-1	324	210	20800	41600	50
xc7a35ticsg324-1L	324	210	20800	41600	50

USING VIVADO + FPGA DEVELOPMENT BOARD(3)

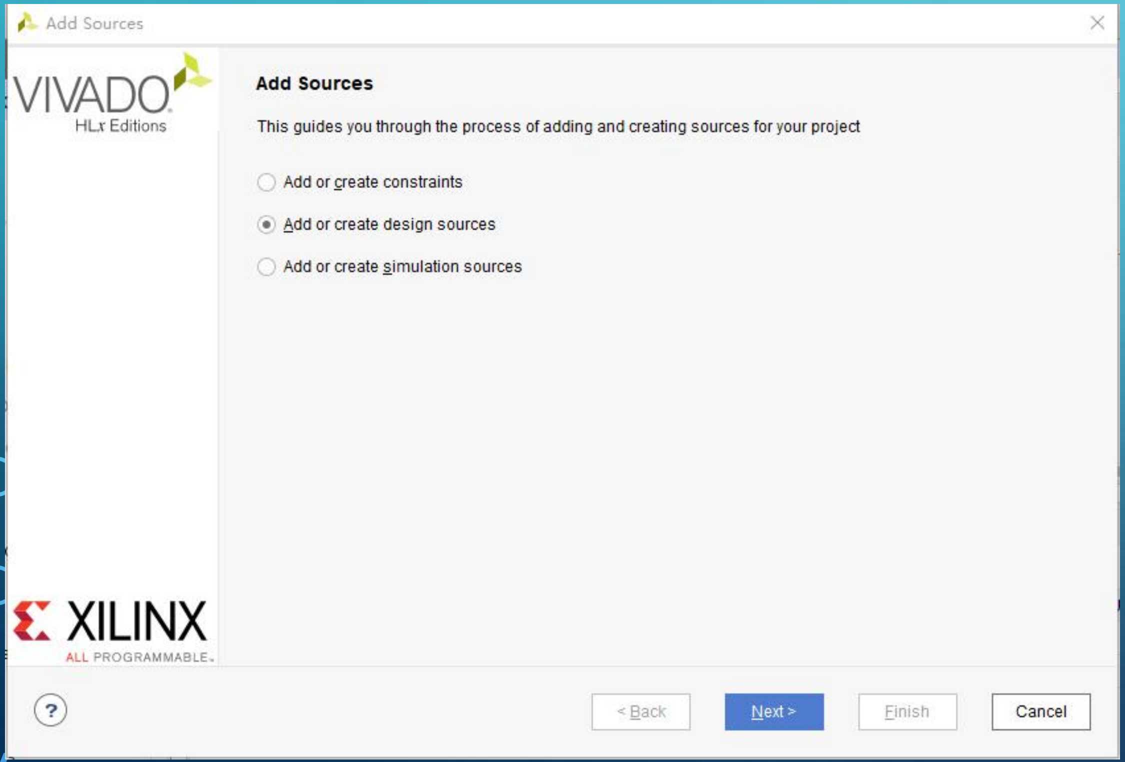
STEP2. Adding source file(s), including **design** file(s), **simulation** file(s) and **constraints** file(s). There are **two ways** to add file(s):

1) Adding file(s) **while creating project**(as shown in the left figure below).

TIPS: If there‘s NO file, you can cread file(s) or just skip the adding files while creating the project.

2) Adding file(s) **after the project is created**(as shown in the right figure below).

Attention: In both ways, You should first select the file type then add or creat a file.



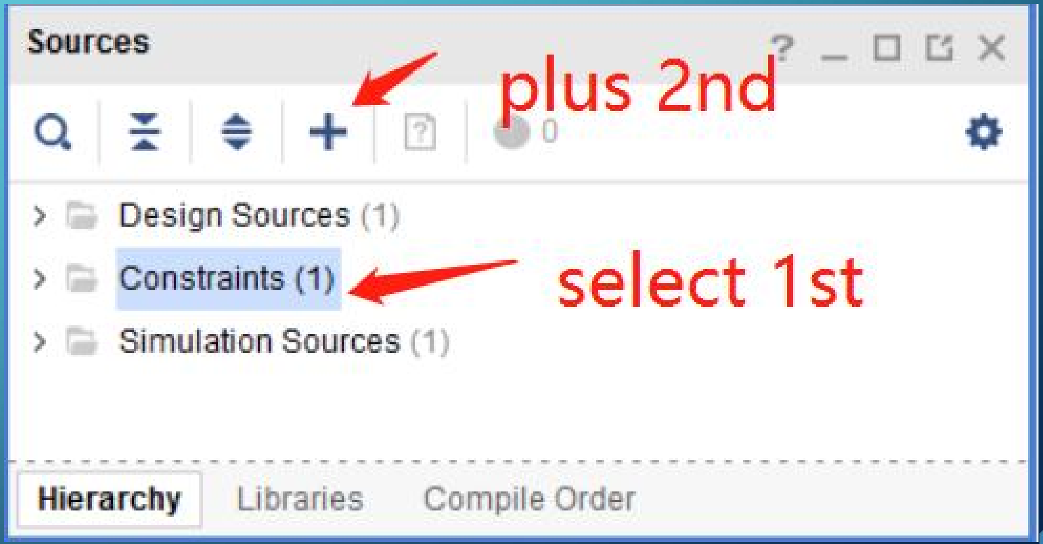
Add Sources

This guides you through the process of adding and creating sources for your project

○ Add or create constraints

● Add or create design sources

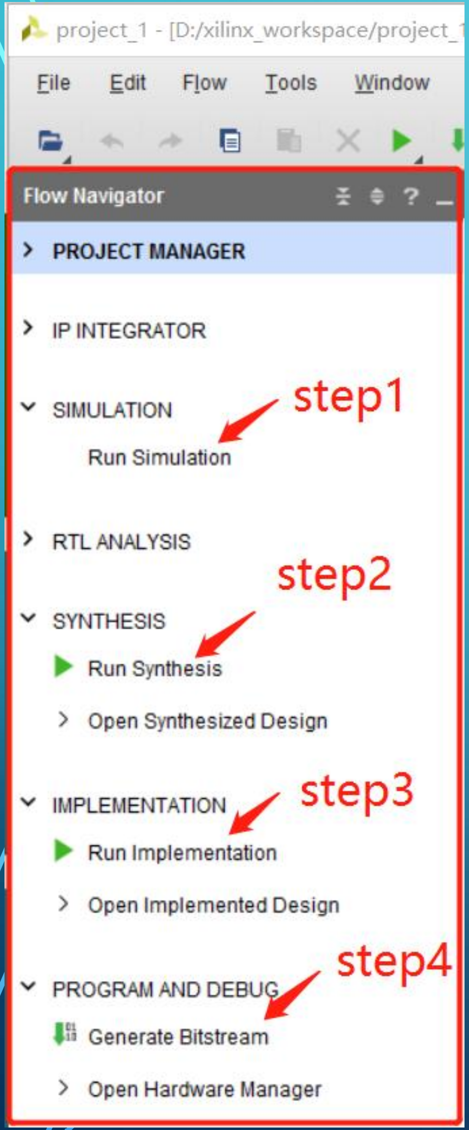
○ Add or create simulation sources



plus 2nd

select 1st

USING VIVADO + FPGA DEVELOPMENT BOARD(4)

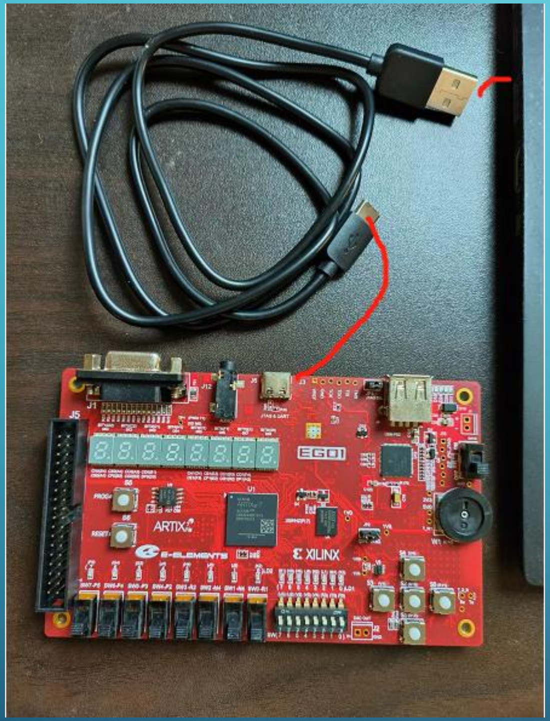


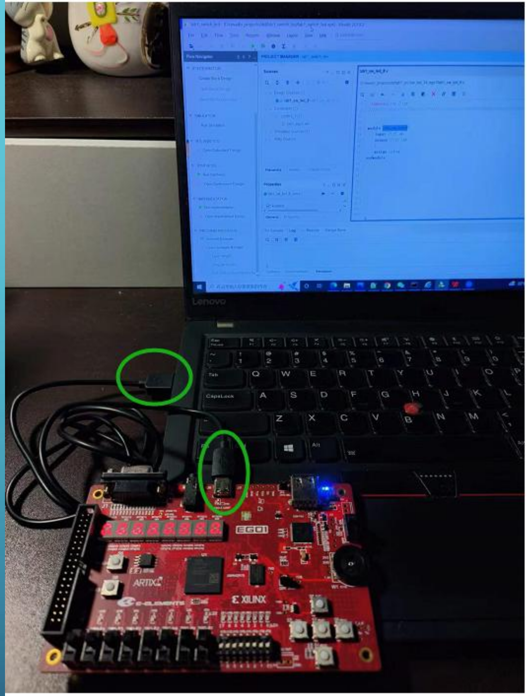
STEP3. Following the steps to **verify the function** of your designed circuit(**step1 in the left figure)** by simulation and generate bitstream file which is used to program FPGA chip(**step4 in the left figure)**

- Do the **simulation(step1 in the left figure)** to verify the function of the designed Circuit
    - After simulation, there will be a waveform which records the states of circuit's input and output signals.
        - **If the** function of the circuit is NOT ok, you should **modify the design file(s)**, then do the simulation**(step1 in the left figure)** again the verigy the function of circuit. Subsequent steps can ONLY be started after the function verification is passed
        - If the function of the circuit is ok , "**run synthesis**"(**step2 in the left figure**) ,then "**run implements**"(**step3 in the left figure)**
- After the implementation is finished, do the "**Generate Bitstream**"(**Step4 in the left figure**) to generate **a bitstream file**(with "**.bit**" as its suffix) .

USING VIVADO + FPGA DEVELOPMENT BOARD(5)

STEP4. First Connect EGO1/Minisys board with PC which runs the Vivado project, then turn on the EGO1/Minisys board.





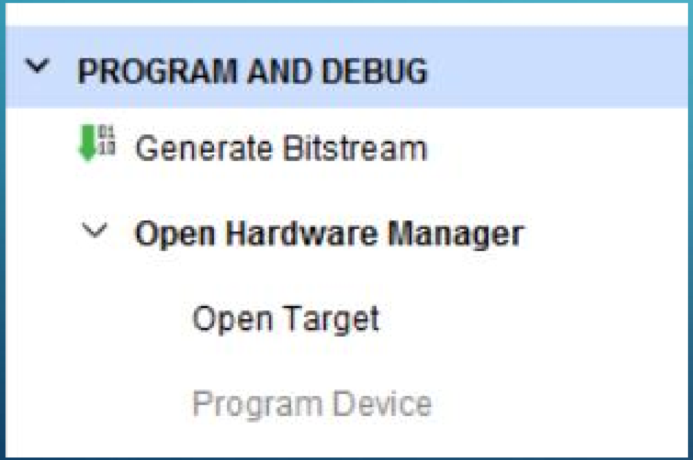
TIPS：
 Here is a demo of connection between EGO1 board and the PC which runs the vivado projet.

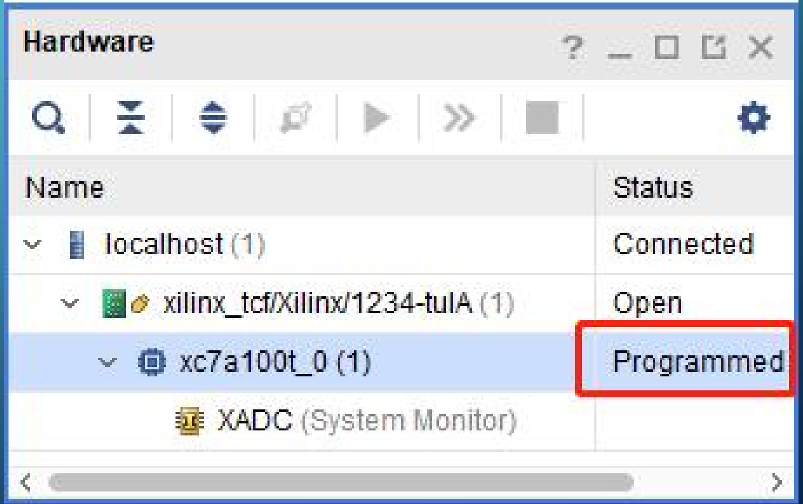
- USB **TypeC** interface for EGO1/Minisys(new version) board
- USB **JTAG** interface for Minisys(old version) board

USING VIVADO + FPGA DEVELOPMENT BOARD(6)

STEP5. First Click “Open Hardware Manager”, then click“**open Target** ” (as shown in the left figure below) to **connect the Vivado project with FPGA chip** which is embeded in the EGO1/Minisys board.

Attention: If the FPGA chip is found by the Vivado project, the chip type could be found in the “**Hardware**” window(as shown in the right figure below, the chip type is xc7a100t), if there is no info about the chip in the “**Hardware**” window, you should redo the STEP4 on last page and the STEP5 on this page untile the FPGA chip is found.

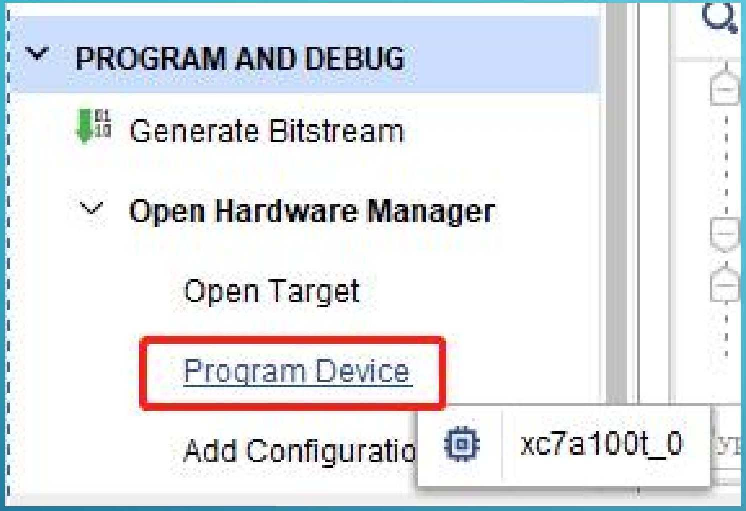


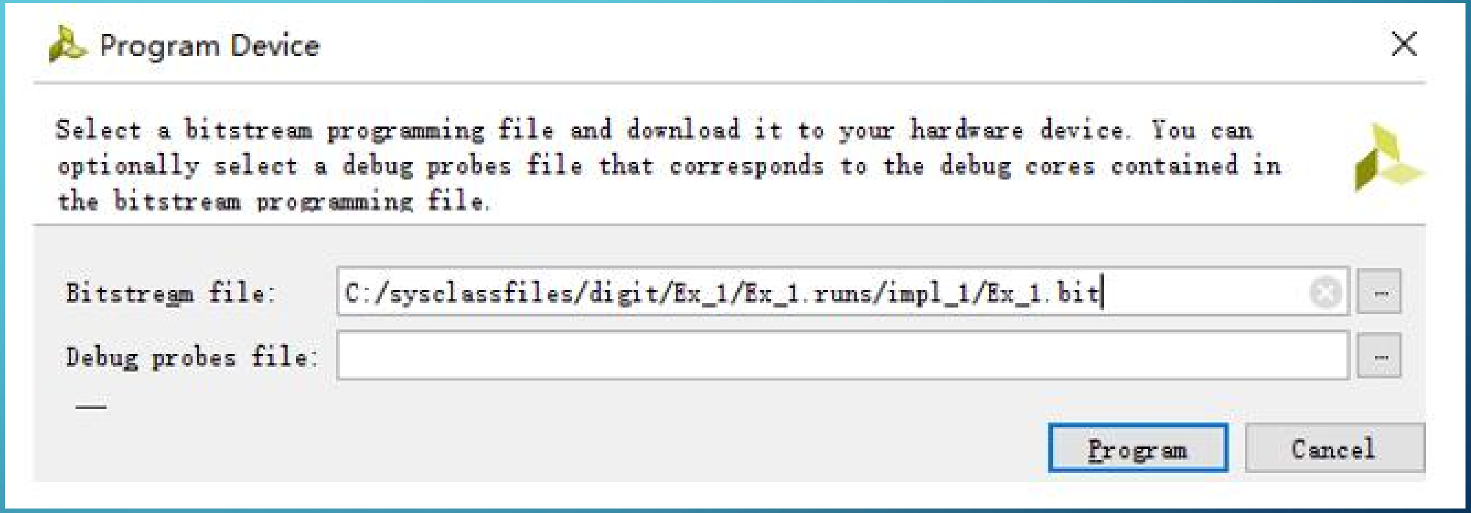


Name	Status
localhost (1)	Connected
xilinx_tcf/Xilinx/1234-tulA (1)	Open
xc7a100t_0 (1)	Programmed
XADC (System Monitor)	

USING VIVADO + FPGA DEVELOPMENT BOARD(7)

STEP6. right click “**Program Device**”, then choose the device name(as shown in the left figure below), select the **bitstream file**(with “.bit” as its suffix) , click “P**rogram**” button(as shown in the right figure below**).**




Program Device

Select a bitstream programming file and download it to your hardware device. You can optionally select a debug probes file that corresponds to the debug cores contained in the bitstream programming file.

Bitstream file: C:/sysclassfiles/digit/Ex_1/Ex_1.runs/impl_1/Ex_1.bit

Debug probes file:

while the the led of “Done” on Minisys/EGO1 is on, it means the bit file is written into the device, it means your circuit is implemented on the FPGA chip, Congratulations!!

Is your circuit functioning properly ? Testing it on the EGO1/Minisys.

A 8-INPUTS-8-OUTPUTS CIRCUIT ON EGO1

lab1_sw_led_8.v

```
module lab1_sw_led_8(
    input [7:0] sw,
    output [7:0] led
    );
    assign led=sw;
endmodule
```

lab1_sw_led_8_sim.v

```
`timescale 1ns / 1ps

module lab1_sw_led_8_sim( );

reg [7:0] tb_sw=24'h000000;
wire [7:0] tb_led;

lab1_sw_led_8 usrc1(
.sw(tb_sw),
.led(tb_led)
);

always #10 tb_sw=tb_sw+1;
endmodule
```

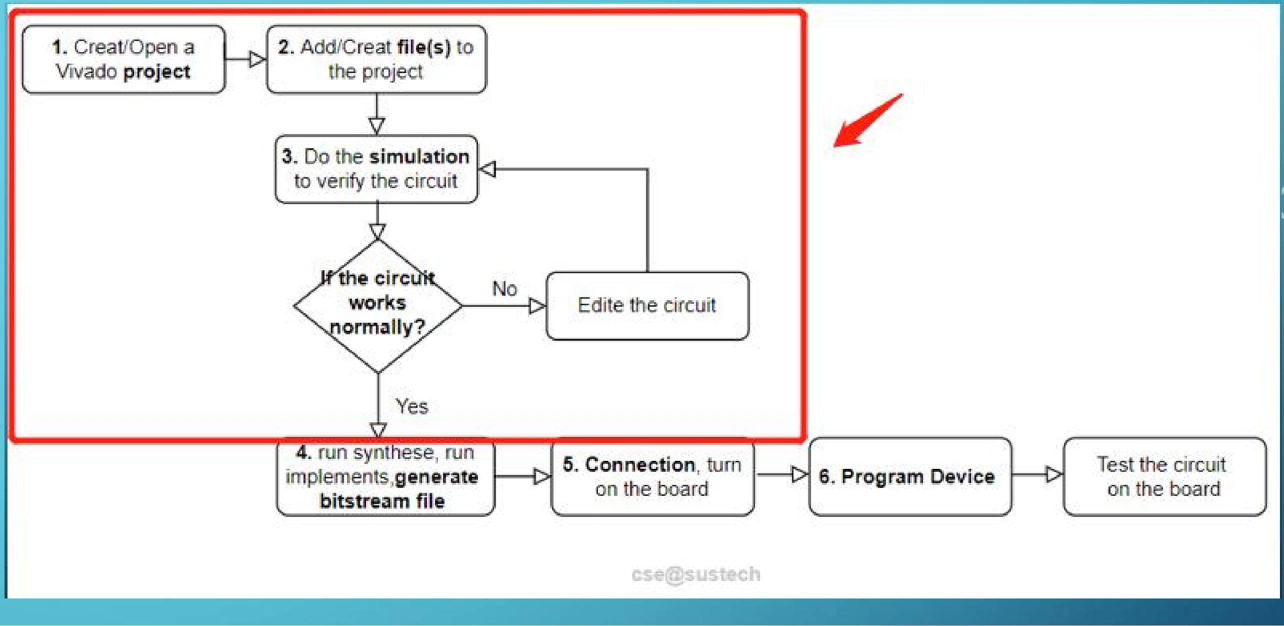
lab1_ego1.xdc

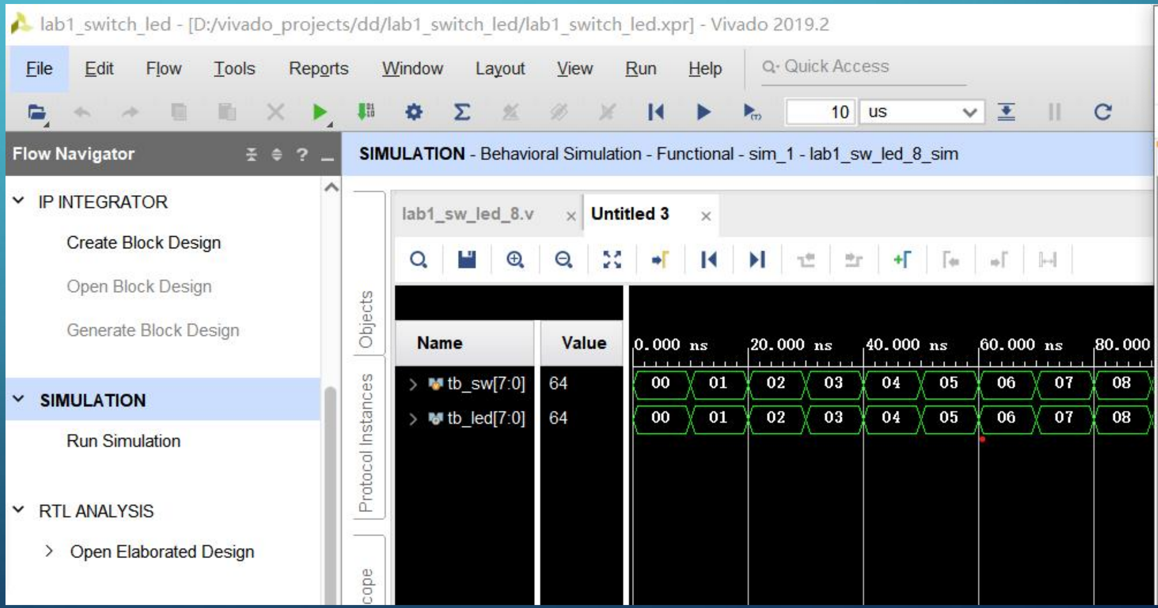
```
set_property IOSTANDARD LVCMOS33 [get_ports {led[7]}]
…
set_property IOSTANDARD LVCMOS33 [get_ports {led[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sw[7]}]
…
set_property IOSTANDARD LVCMOS33 [get_ports {sw[0]}]
set_property PACKAGE_PIN F6 [get_ports {led[7]}]
…
set_property PACKAGE_PIN K2 [get_ports {led[0]}]
set_property PACKAGE_PIN P5 [get_ports {sw[7]}]
…
set_property PACKAGE_PIN R1 [get_ports {sw[0]}]
```

Q: If “lab1_sw_led_8_sim.v” is removed from the Vivado project, will the circuit on the FPGA chip work or not?

PRACTICE1

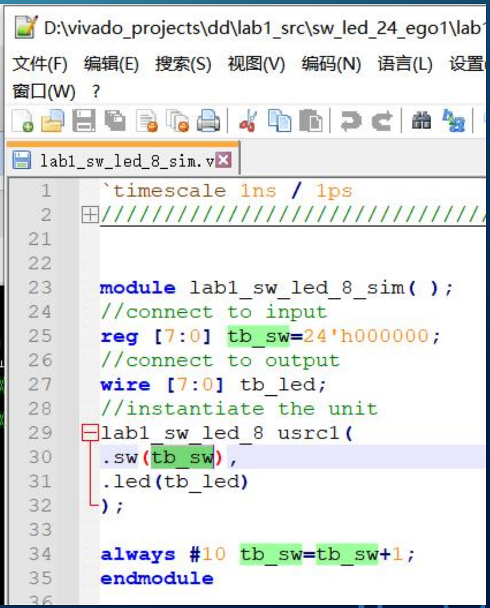
- If you have NOT got the board(with FPGA chip embeded), do practice1, following the steps shown in the top figure on the right side.
- A waveform would be generated to show the logic relationship between the input and output of the designed circuit.
- Is the logic relationship shown in the waveform same with your design?


1. Creat/Open a Vivado project
2. Add/Creat file(s) to the project
3. Do the simulation to verify the circuit
If the circuit works normally?
No
Edite the circuit
Yes
4. run synthese, run implements,generate bitstream file
5. Connection, turn on the board
6. Program Device
Test the circuit on the board
cse@sustech



SIMULATION - Behavioral Simulation - Functional - sim_1 - lab1_sw_led_8_sim

```
Name          Value   0.000 ns    20.000 ns    40.000 ns    60.000 ns    80.000
> tb_sw[7:0]   64     00    01    02    03    04    05    06    07    08
> tb_led[7:0]  64     00    01    02    03    04    05    06    07    08
```



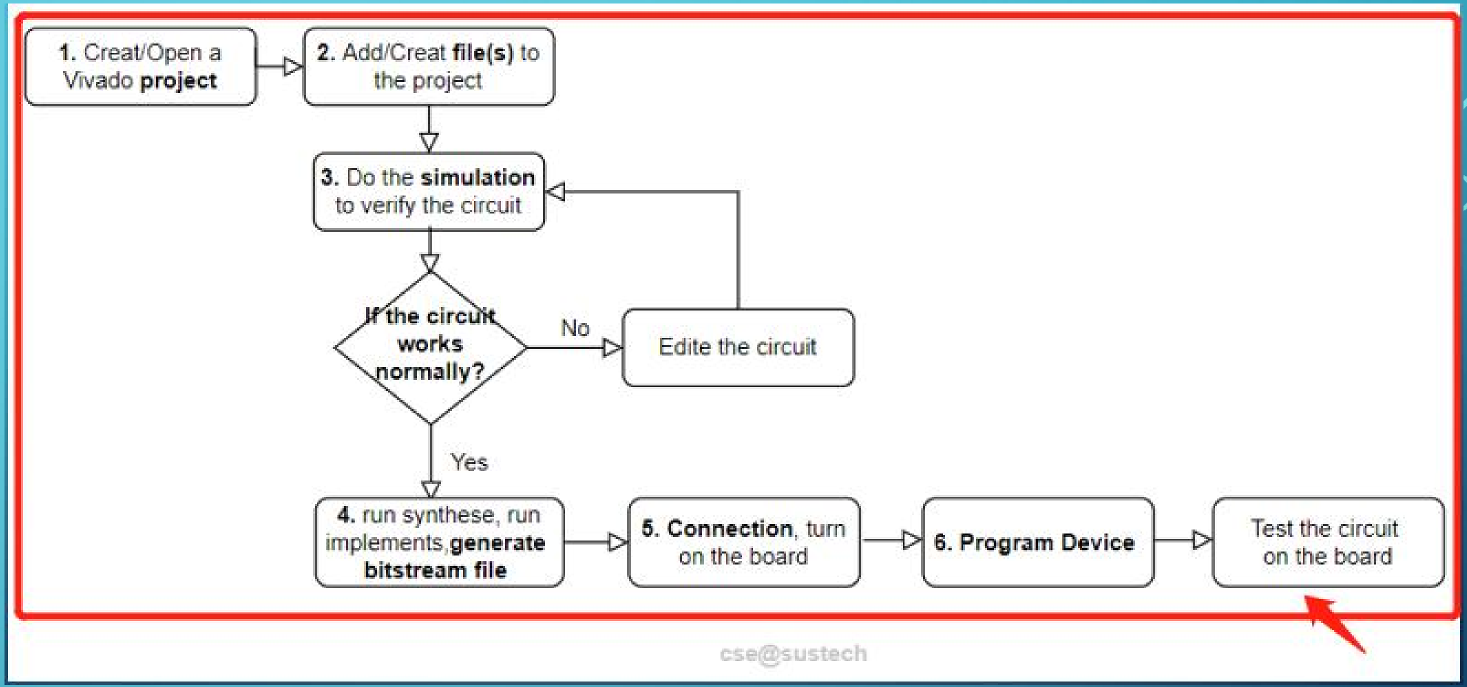
```
`timescale 1ns / 1ps
//////////////////////////////

module lab1_sw_led_8_sim( );
//connect to input
reg [7:0] tb_sw=24'h000000;
//connect to output
wire [7:0] tb_led;
//instantiate the unit
lab1_sw_led_8 usrc1(
.sw(tb_sw),
.led(tb_led)
);

always #10 tb_sw=tb_sw+1;
endmodule
```

PRACTICE2

- If you HAVE got the board(with FPGA chip embeded), do practice2, following the steps shown in the top figure on the right side.
- Test the designed circuit on the board, change the input(here is the dail switch of EGO1) to absert to state of output(here is the led of EGO1).
- Is the logic relationship between the input and the output same with your design?



1. Creat/Open a Vivado **project**

2. Add/Creat **file(s)** to the project

3. Do the **simulation** to verify the circuit

If the circuit works normally?

No

Edite the circuit

Yes

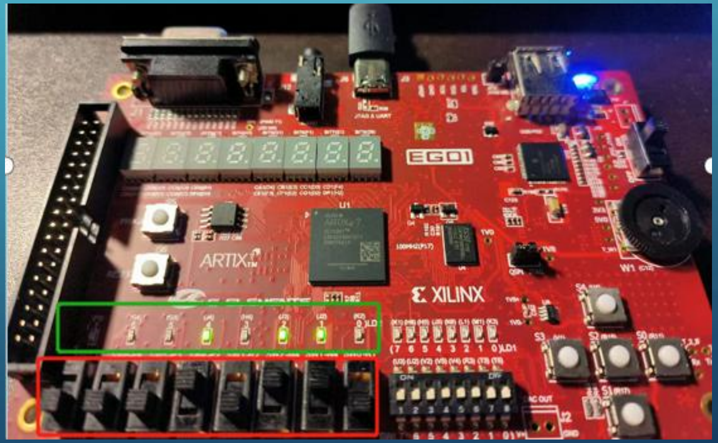
4. run synthese, run implements,**generate bitstream file**

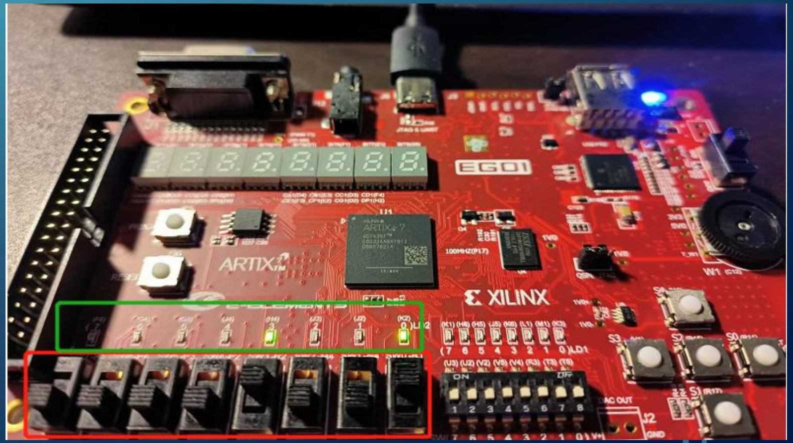
5. Connection, turn on the board

6. Program Device

Test the circuit on the board

cse@sustech





A 24-INPUTS-24-OUTPUTS CIRCUIT ON MINISYS

lab1_sw_led_24.v

lab1_sw_led_24_sim.v

lab1_minisys.xdc

Q: If a 12-inputs-12-outputs circuit is designed to work on Minisys board, which of the above files need to be modified to make the circuit work on the Minsys board?

TIPS:
while using Minisys(new version) board/EGO1 board, connect its typeC interface to the computer which run Vivado project by typeC USB wire.

while using Minisys(old version) board, connect its USB-Jtag interface to the computer which run Vivado project by typeB USB wire.