

# FPGA Emulation

Hsi-Pin Ma

<https://eiclass.nthu.edu.tw/course/18498>

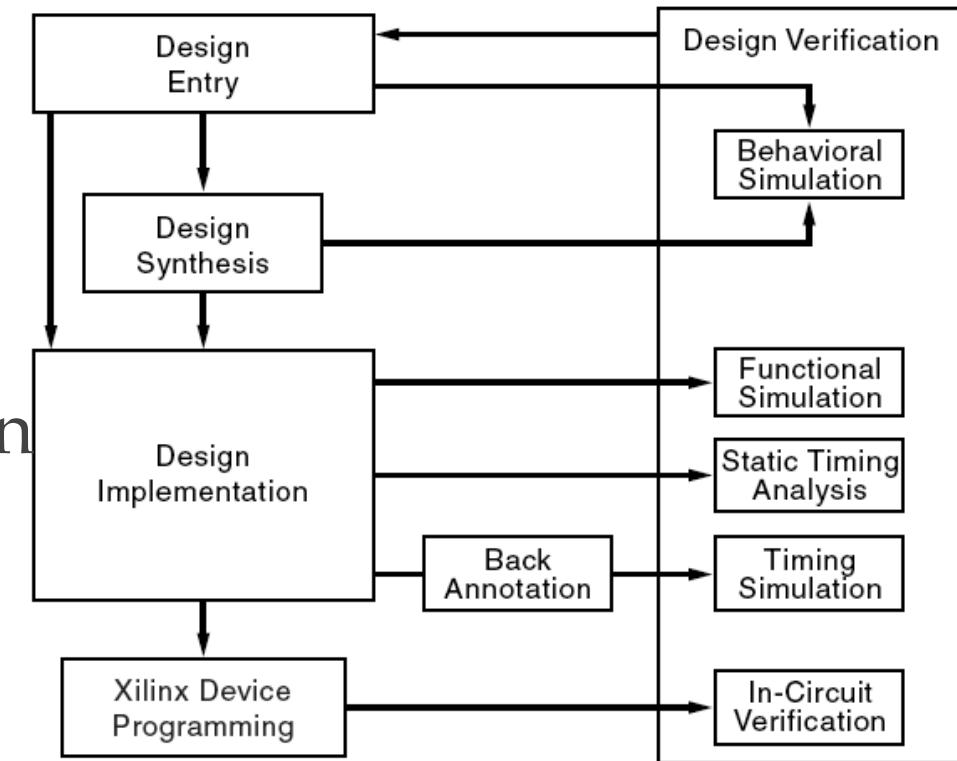
Department of Electrical Engineering  
National Tsing Hua University

# Notes from Lab1

- Always 'SAVE' before next step
- Select the right file for the next step
  - For simulation
  - For implementation
- Do not use specific names for files and directories
  - Leading with number,
  - Names with space,
  - Names in Chinese

# Design Flow

- General design flow
  - Design construction
  - Behavioral simulation
  - Design implementation
  - Timing simulation
- HDL-based design Flow



# Important Notes

- **Draw schematic first** and then construct Verilog codes.
- Verilog RTL coding philosophy is not the same as C programming
  - Every Verilog RTL construct has its own logic mapping (for synthesis)
  - You should have the logics (draw schematic) first and then the RTL codes
  - You have to write **synthesizable** RTL codes

# Digilent Basys 3 Demo Board

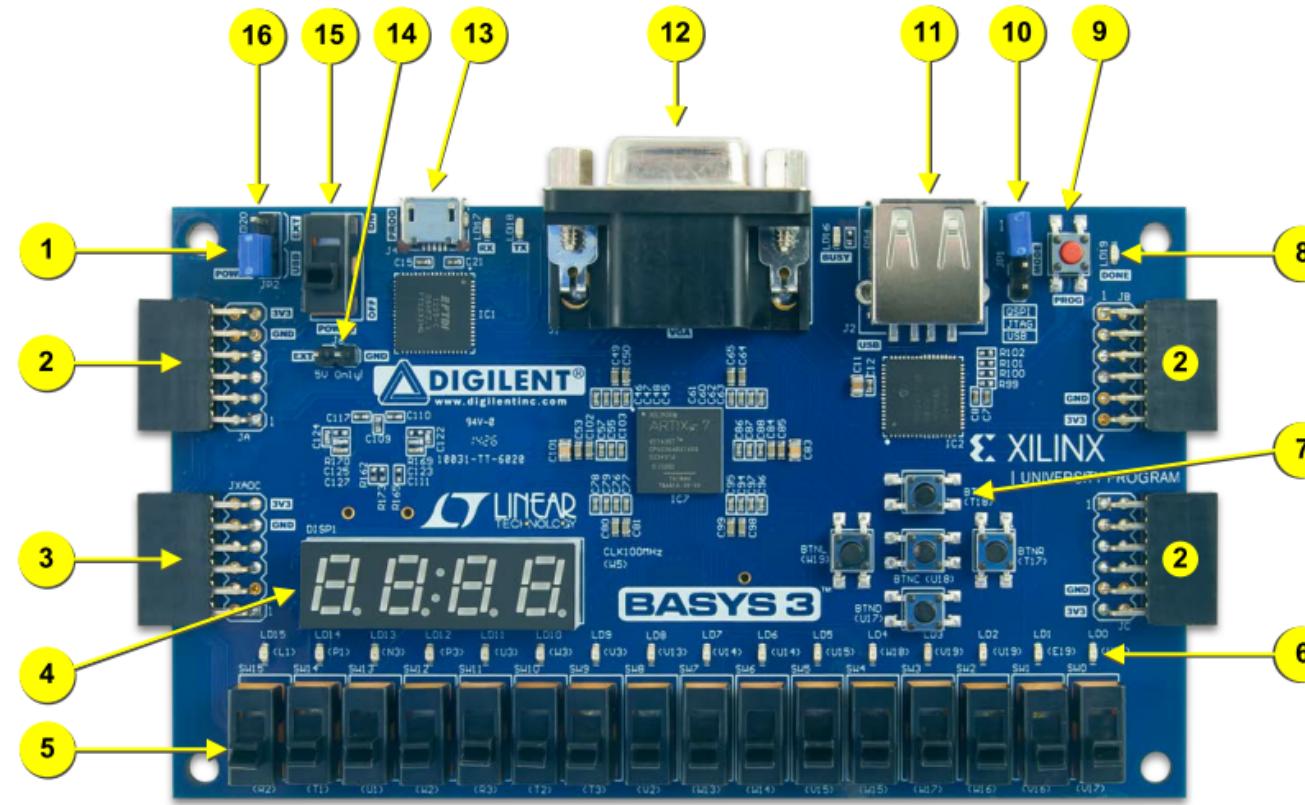


Figure 1. Basys3 FPGA board with callouts.

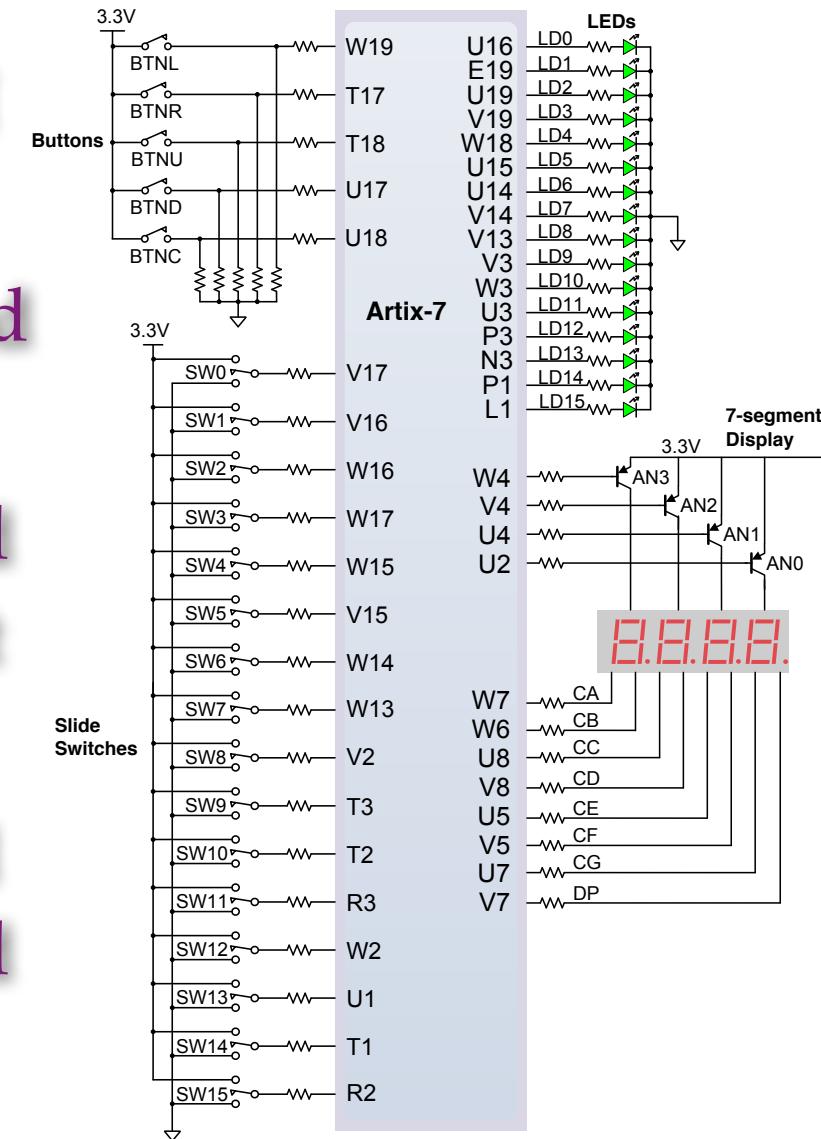
Callout	Component Description
1	Power good LED
2	Pmod connector(s)
3	Analog signal Pmod connector (XADC)
4	Four digit 7-segment display
5	Slide switches (16)
6	LEDs (16)
7	Pushbuttons (5)
8	FPGA programming done LED
9	FPGA configuration reset button
10	Programming mode jumper
11	USB host connector
12	VGA connector
13	Shared UART/ JTAG USB port
14	External power connector
15	Power Switch
16	Power Select Jumper

<https://reference.digilentinc.com/reference/programmable-logic/basys-3/start?redirect=1>

Check Basys 3 board reference manual for details

# Input/Output Connections

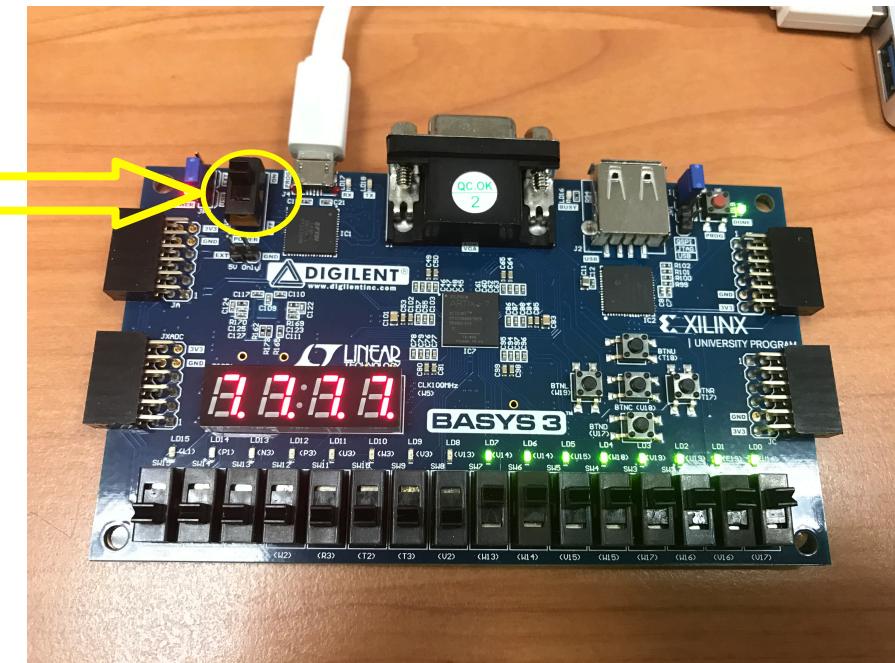
- LEDs are pre-wired **HIGH** active control
- 7-Seg Display are pre-wired **LOW** active control
- Push buttons are pre-wired **LOW** when they are at rest
- DIP switches generate **HIGH** when tuned up and generate **LOW** when tuned down



# Test Your FPGA Board (1/3)

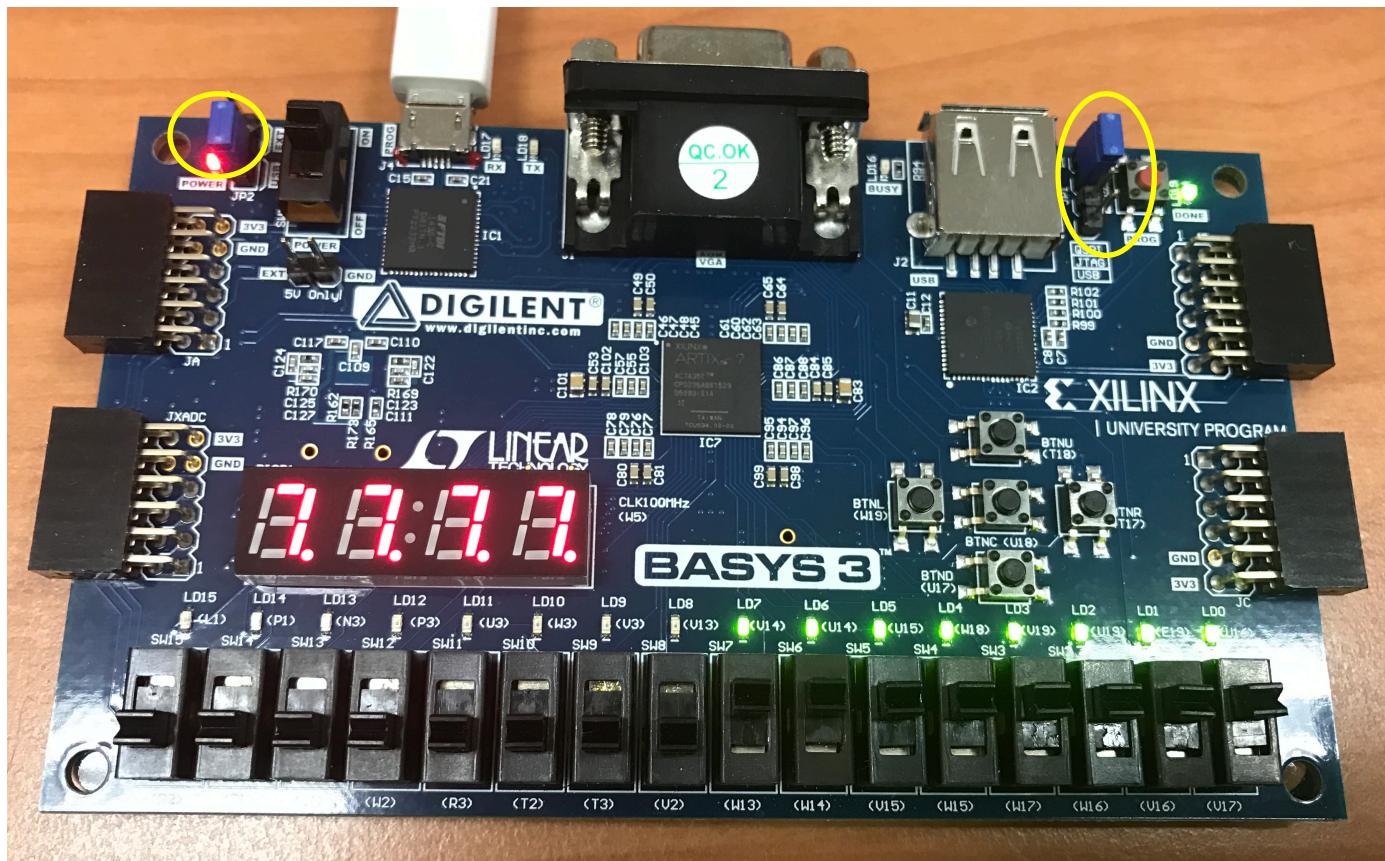
- Connect the demo board to PC and also the power supply

- 所有的線接好再開電源
- 關閉電源後再拔所有的線
- 勿用導體接觸針腳
- 插座別插反了
- 避免長時間開機



# Test Your FPGA Board (2/3)

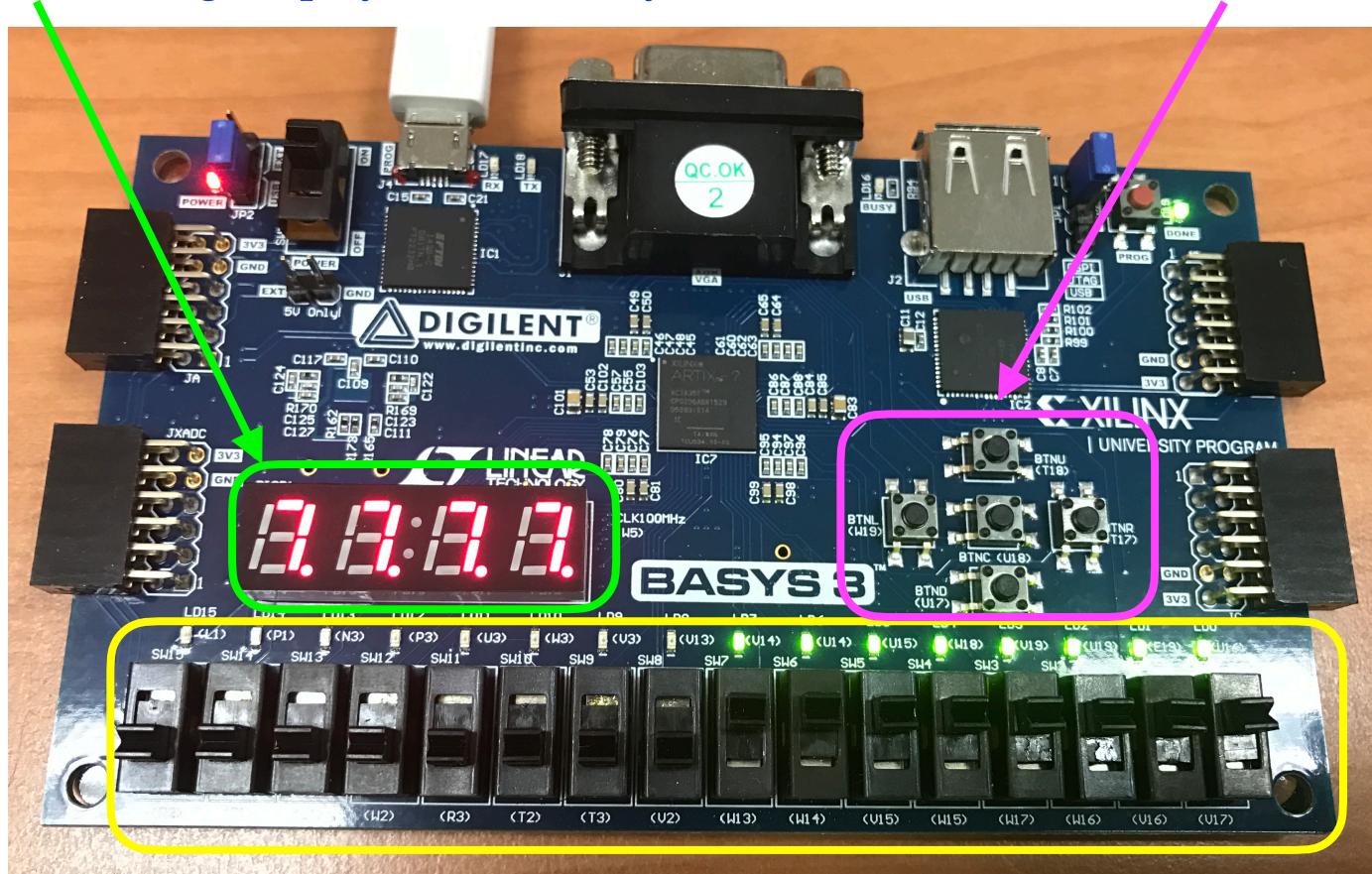
- Don't move or remove the jumpers



# Test Your FPGA Board (3/3)

BCD counts on 7-Seg Displays continuously

Push buttons control the 7-Seg display



DIP Switch control the LEDs

# FPGA Emulation Using Xilinx Vivado

# Design Flow

- Design Source Preparation
  - Design modules (.v)
  - Design constraints (I/O pin assignments) (.xdc)
- Design Simulation
  - testbench (.v)
- Design Synthesis and Implementation

# Create New Project (2/3)

Continue from previous unit slides

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: cpg236

Reset All Filters

Search: xc7a35tcp (1 match)

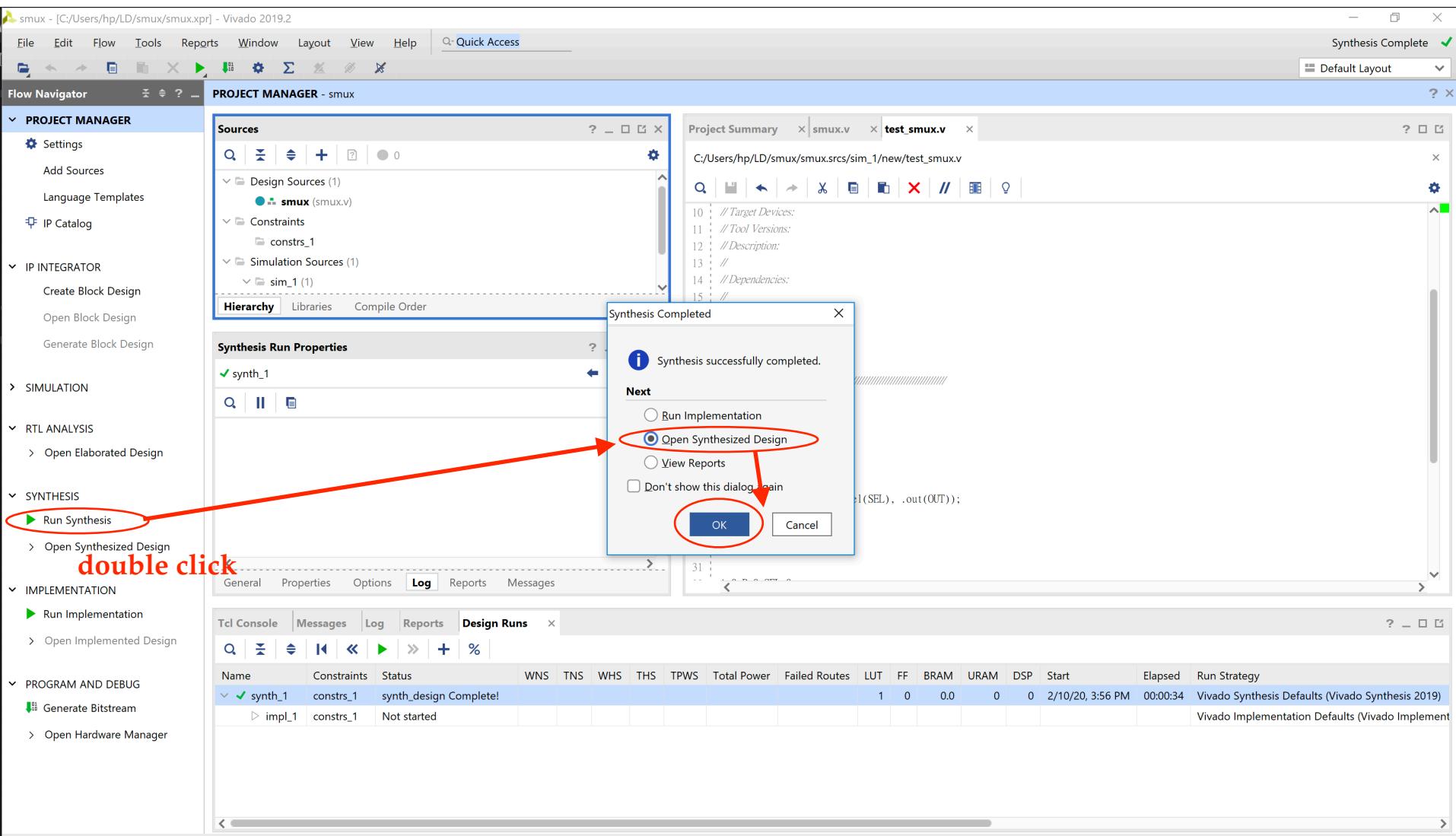
Part	I/O Pin Count	Block RAMs	DSPs	FlipFlops	GTPE2 Transceivers	Gb Transceivers	Available IOBs	LUT Elements
xc7a35tcp	236	50	90	41600	2	2	106	20800

?

< Back **Next >** Finish Cancel

Part	I/O Pin Count	Block RAMs	DSPs	FlipFlops	GTPE2 Transceivers	Gb Transceivers	Available IOBs	LUT Elements
xc7a35tcp	236	50	90	41600	2	2	106	20800

# I/O Pins Assignment (1/4)



# I/O Pins Assignment (2/4)

3

Remember to save your assignment

1

The screenshot shows the Vivado 2019.2 interface with several windows open:

- Flow Navigator**: A sidebar on the left with a red circle around the 'File' icon.
- SYNTHESIZED DESIGN**: The main workspace showing the 'Device Constraints' tab.
- I/O Port Properties**: A floating window for the 'out' port, showing details like Name: out, Direction: OUT, Package pin: U16, Site type: IO\_L23N\_T3\_A02\_D18\_14, and Cell: out\_OBUF\_inst.
- Package**: A window showing the physical layout of the device with various I/O pins and internal components.
- Save Constraints**: A dialog box for saving constraints.
  - Create a new file** radio button selected.
  - File type:** XDC (selected from dropdown).
  - File name:** smux.xdc (highlighted with a red circle).
  - File location:** <Local to Project> (dropdown).
  - Select an existing file** radio button (disabled).
  - select a target file** dropdown (disabled).
  - OK** and **Cancel** buttons.

Annotations with numbers:

- Red circle highlights the 'File' icon in the Flow Navigator.
- Red circle highlights the 'I/O Ports' tab in the synthesized design window.
- Blue box highlights the 'File' icon in the Save Constraints dialog.
- Blue box highlights the 'File name:' field in the Save Constraints dialog.

Text annotations:

- change to LVCMOS33** (red text) points to the 'I/O Std' column in the I/O Port Properties table.
- type the constraint file name** (red text) is inside the Save Constraints dialog.

Name	Direction	Interface	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength
a	IN			V17	<input checked="" type="checkbox"/>	14	LVCMOS33*	-	3.300	
b	IN			V16	<input checked="" type="checkbox"/>	14	LVCMOS33*	-	3.300	
out	OUT			U16	<input checked="" type="checkbox"/>	14	LVCMOS33*	-	3.300	12
sel	IN			W16	<input checked="" type="checkbox"/>	14	LVCMOS33*	-	3.300	

# I/O Pins Assignment (3/4)

smux - [C:/Users/hp/LD/smux/smux.xpr] - Vivado 2019.2

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

Synthesis Out-of-date details I/O Planning

Flow Navigator

**PROJECT MANAGER**

- Settings
- Add Sources
- Language Templates
- IP Catalog

**IP INTEGRATOR**

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

**SIMULATION**

**RTL ANALYSIS**

- Open Elaborated Design

**SYNTHESIS**

- Run Synthesis
- Open Synthesized Design**
- Constraints Wizard
- Edit Timing Constraints
- Set Up Debug
- Report Timing Summary
- Report Clock Networks
- Report Clock Interaction
- Report Methodology
- Report DRC
- Report Noise
- Report Utilization

**SYNTHESIZED DESIGN - xc7a35tcpg236-1**

Sources Netlist Device Constraints

Internal VREF

- 0.6V
- 0.675V
- 0.75V
- 0.9V

Drop I/O banks on voltages or the "NONE" folder to set/unset Internal VREF.

**I/O Port Properties**

Name: out  
Direction: OUT  
Package pin: U16  Fixed  
Site type: IO\_L23N\_T3\_A02\_D18\_14  
Site info: IOB\_X0Y3  
Cell: out\_OBUF\_inst

OK Cancel

Confirm Close

OK to close 'Synthesized Design'?  
 Don't show this dialog again.

OK

Package Device smux.v test\_smux.v

**Tcl Console Messages Log Reports Design Runs Package Pins I/O Ports**

Name	Direction	Interface	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type	Off-Chip Termination	IN_TERM
a	IN			V17	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300			NONE	<input checked="" type="checkbox"/>	NONE	N
b	IN			V16	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300			NONE	<input checked="" type="checkbox"/>	NONE	N
out	OUT			U16	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300	12	SLOW	<input checked="" type="checkbox"/>	NONE	FP_VTT_50	N
sel	IN			W16	<input checked="" type="checkbox"/>	14	LVC MOS33*	3.300			NONE	<input checked="" type="checkbox"/>	NONE	N

# I/O Pins Assignment (4/4)

The screenshot shows the Vivado 2019.2 interface with the project 'smux' open. The 'PROJECT MANAGER' pane on the left lists various project management options like Settings, Add Sources, Language Templates, and IP Catalog. The 'SIMULATION' and 'SYNTHESIS' sections are also visible. The central area shows the 'PROJECT MANAGER - smux' window with the 'Sources' tab selected. Under 'Design Sources', there is a file named 'smux'. Under 'Constraints', there is a folder 'constrs\_1' containing a file 'smux.xdc (target)'. A red circle highlights this file with the text 'double click' overlaid. To the right, the 'Project Summary' window shows the contents of 'smux.xdc':

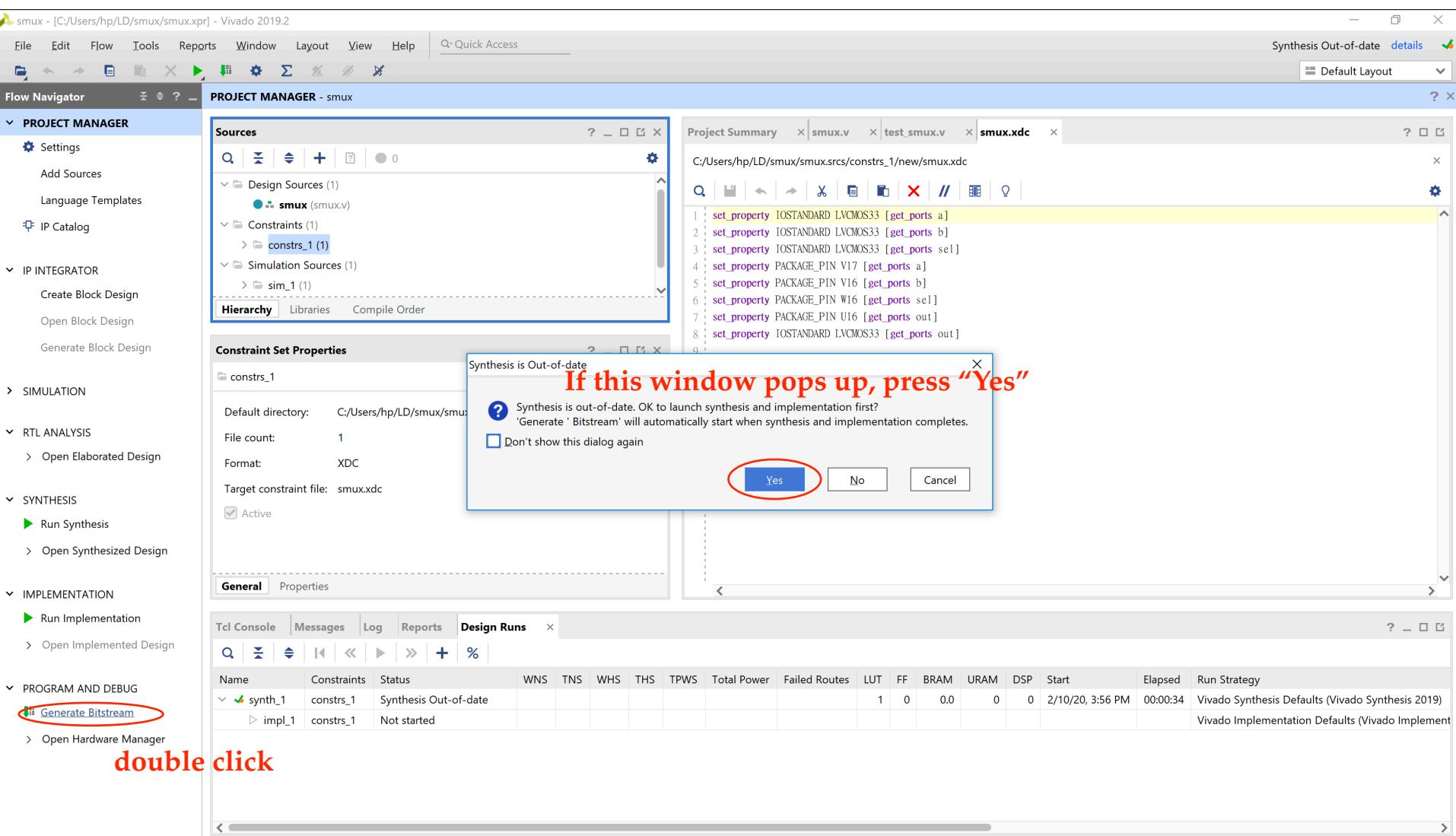
```
C:/Users/hp/LD/smux/smux.srscs/constrs_1/new/smux.xdc
1: set_property IOSTANDARD LVCMOS33 [get_ports a]
2: set_property IOSTANDARD LVCMOS33 [get_ports b]
3: set_property IOSTANDARD LVCMOS33 [get_ports sel]
4: set_property PACKAGE_PIN V17 [get_ports a]
5: set_property PACKAGE_PIN V16 [get_ports b]
6: set_property PACKAGE_PIN W16 [get_ports sel]
7: set_property PACKAGE_PIN U16 [get_ports out]
8: set_property IOSTANDARD LVCMOS33 [get_ports out]
```

Below the project summary, a large red text box contains the instruction: 'Or you can create the file directly and then added by "Add Sources" (constraints)'.

The bottom section of the interface shows the 'Design Runs' tab with a table of synthesis results:

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy
synth_1	constrs_1	Synthesis Out-of-date								1	0	0.0	0	0	2/10/20, 3:56 PM	00:00:34	Vivado Synthesis Defaults (Vivado Synthesis 2019)
impl_1	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implement)

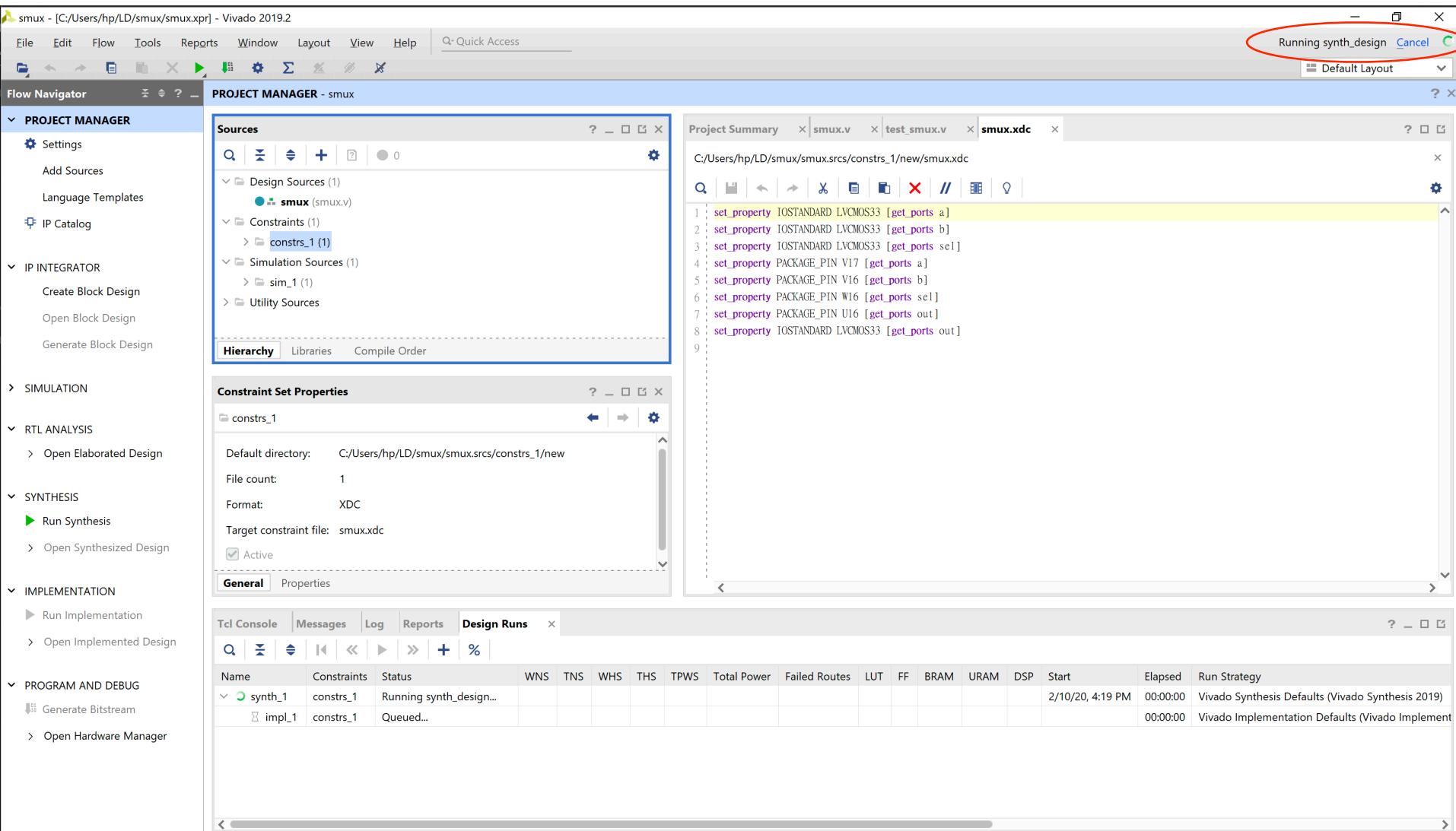
# Synthesis and Implementation (1/7)



Generate a programming file after implementation

# Synthesis and Implementation (2/7)

implementation progress information



The screenshot shows the Vivado 2019.2 interface with the following details:

- Project Manager - smux**: Shows the project structure with sources (smux.v), constraints (constrs\_1), simulation sources (sim\_1), and utility sources.
- Constraint Set Properties - constrs\_1**: Displays properties for the constraint set, including the target constraint file (smux.xdc) and the active status.
- Design Runs**: A table showing the status of design runs. One run, "synth\_1" under "constrs\_1", is listed as "Running synth\_design...".
- Tcl Console**: Shows the command "Running synth\_design...".
- Messages**: Shows the message "Running synth\_design...".
- Progress Bar**: Located at the bottom of the interface, indicating the progress of the current synthesis task.

A red circle highlights the status bar message "Running synth\_design...".

# Synthesis and Implementation (3/7)

1 Connect and power on the FPGA board

2

Bitstream Generation Completed

Bitstream Generation successfully completed.

Next

Open Implemented Design

View Reports

Open Hardware Manager

Generate Memory Configuration File

Don't show this dialog again

OK Cancel

PROJECT MANAGER - smux

Sources

Design Sources (1) smux (smux.v)

Constraints (1) constrs\_1 (1)

Simulation Sources (1) sim\_1 (1)

Utility Sources

Hierarchy Libraries Compile Order

Constraint Set Properties

constrs\_1

Default directory: C:/Users/hp/LD/smux/smux.srscs/constrs\_1/new

File count: 1

Format: XDC

Target constraint file: smux.xdc

Active

General Properties

Tcl Console Messages Log Reports Design Runs

Name Constraints Status WNS TNS WHS THS TPWS Total Power Failed Routes LUT FF BRAM URAM DSP Start Elapsed Run Strategy

synth\_1 constrs\_1 synth\_design Complete! NA NA NA NA NA 1.103 1 0 0.0 0 0 2/10/20, 4:19 PM 00:00:44 Vivado Synthesis Defaults (Vivado Synthesis 2)

impl\_1 constrs\_1 write\_bitstream Complete! NA NA NA NA NA 0 1 0 0.0 0 0 2/10/20, 4:20 PM 00:01:09 Vivado Implementation Defaults (Vivado Impl)

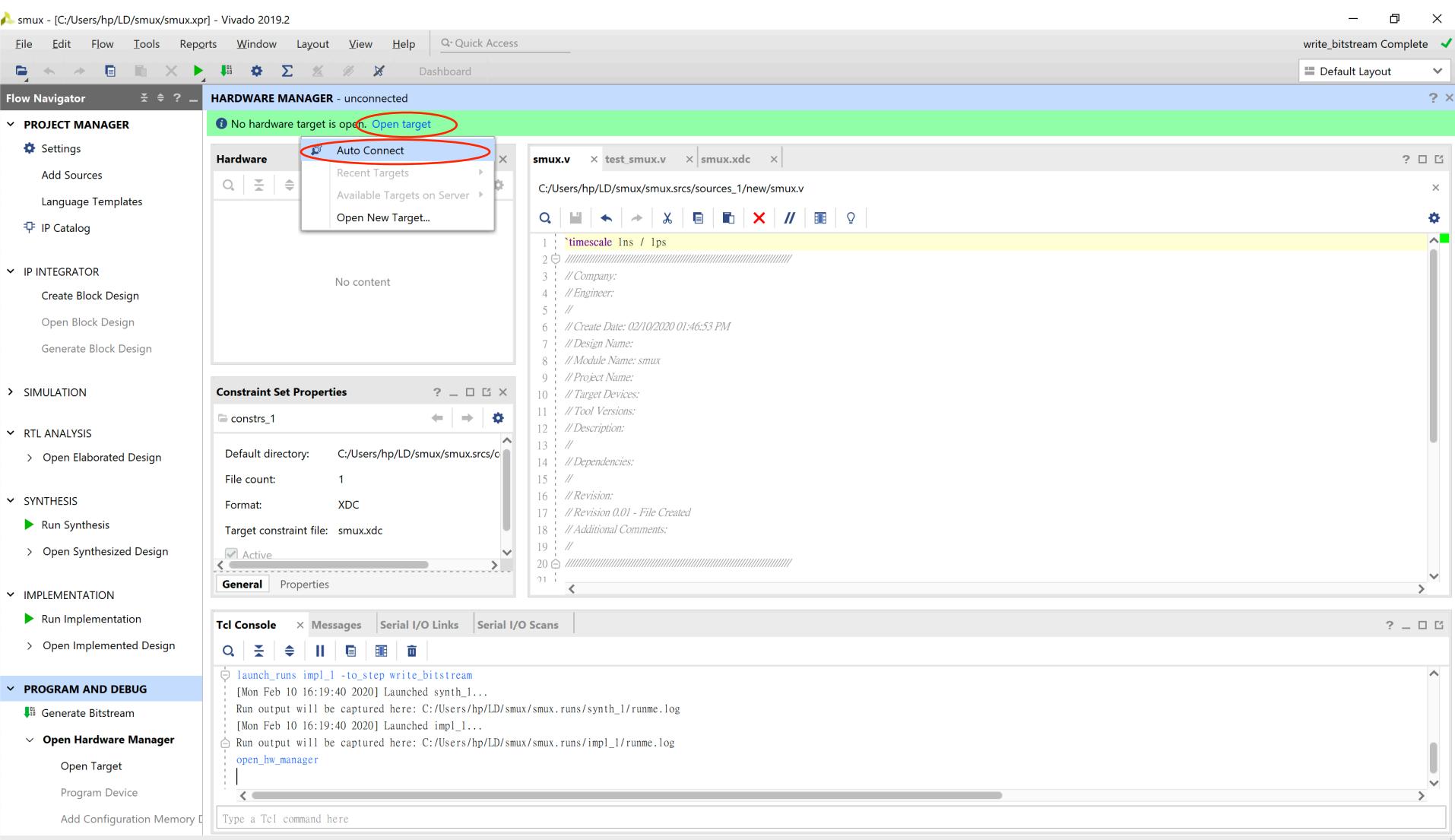
Constraint Set: constrs\_1

After synthesis and implementation

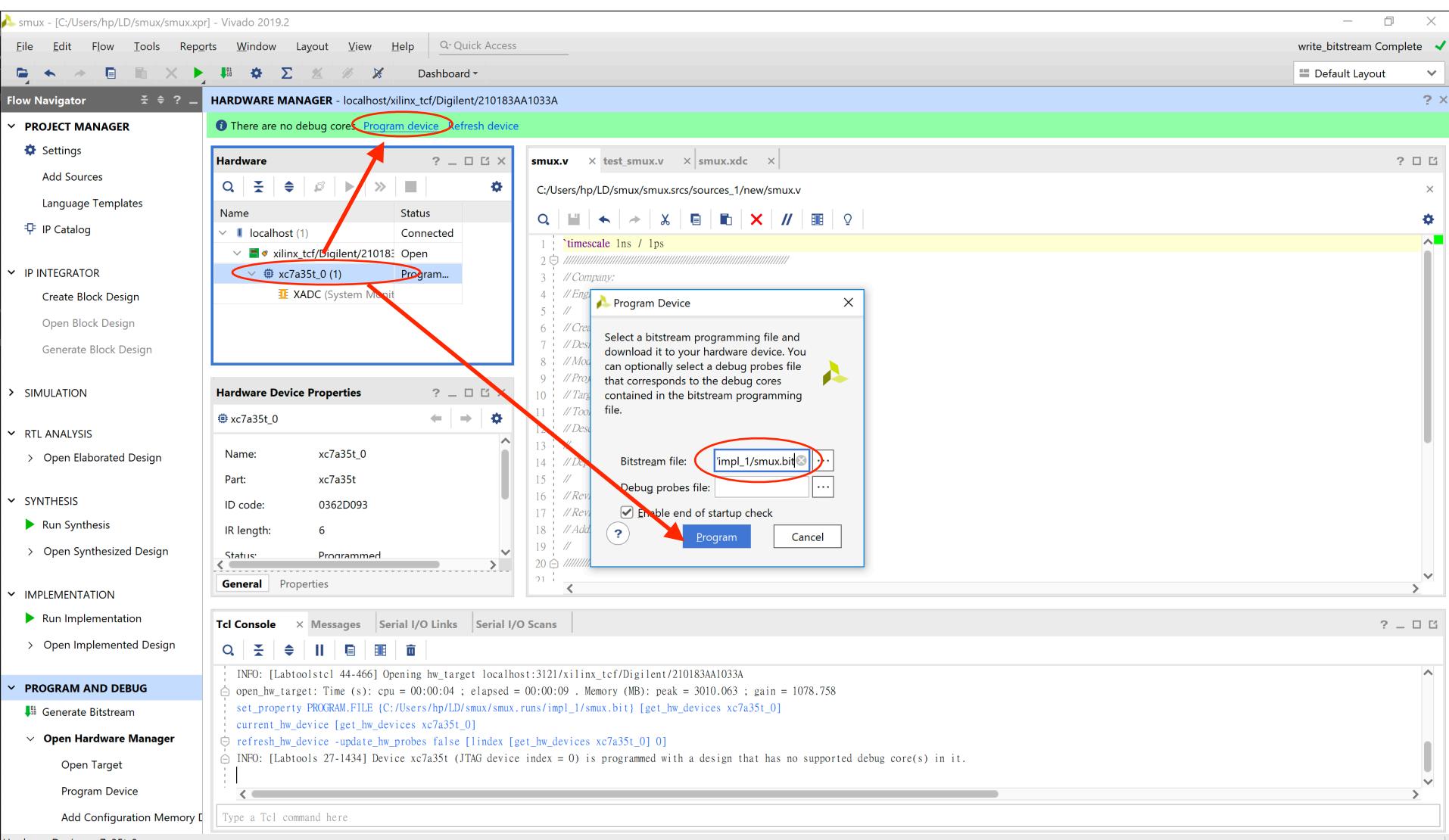
write\_bitstream Complete ✓

Default Layout

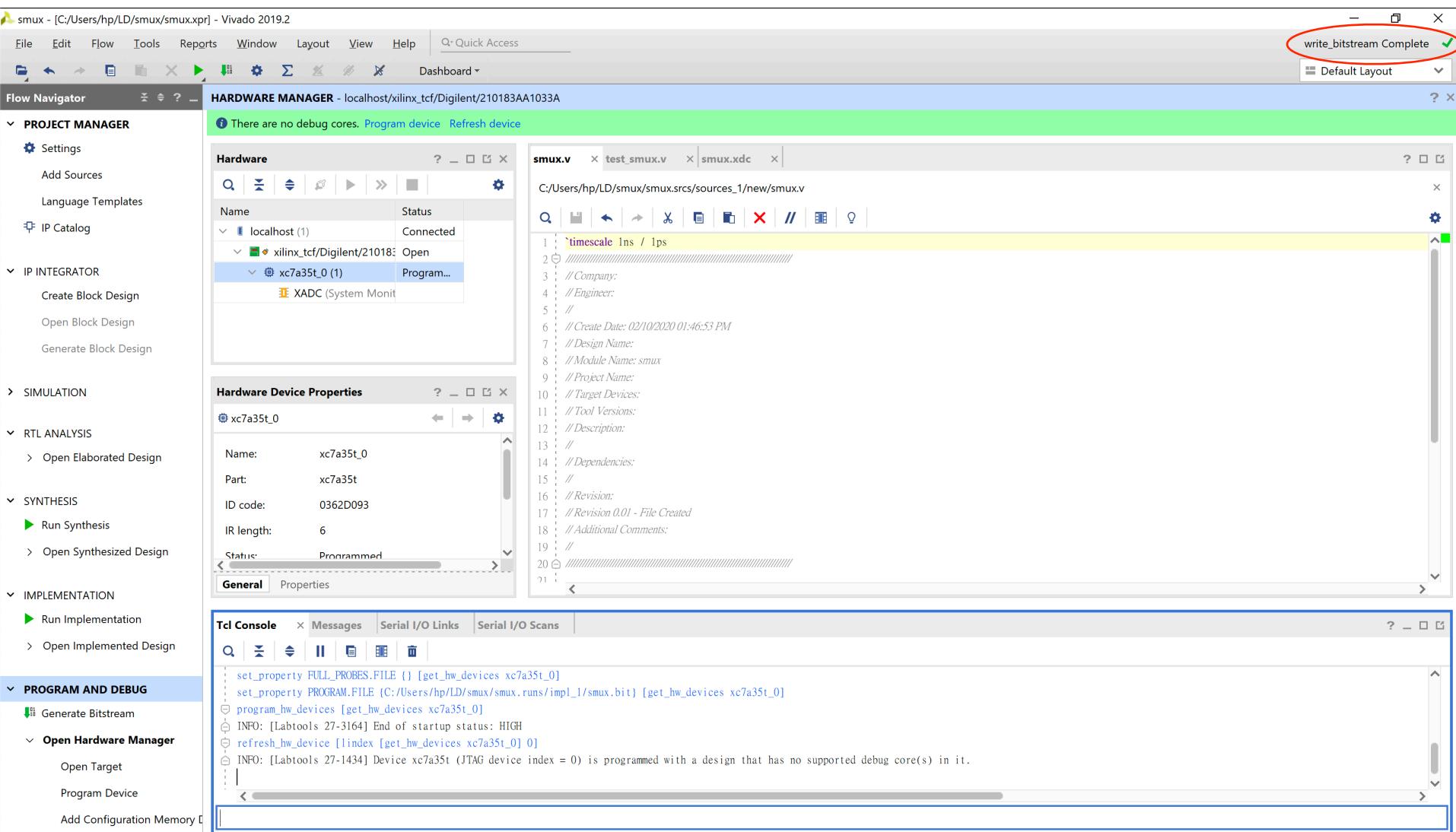
# Synthesis and Implementation (4/7)



# Synthesis and Implementation (5/7)

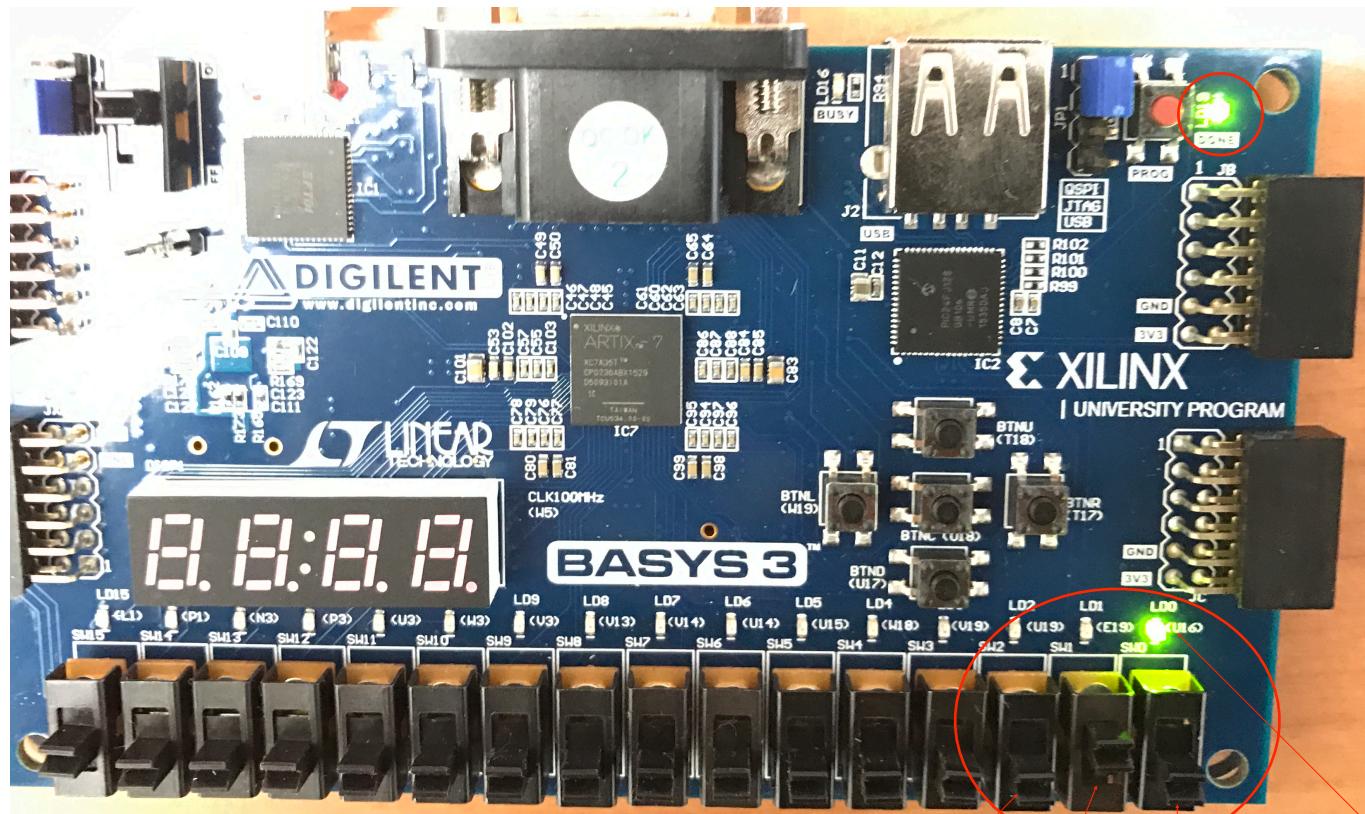


# Synthesis and Implementation (6/7)



# Synthesis and Implementation (7/7)

Program Finished



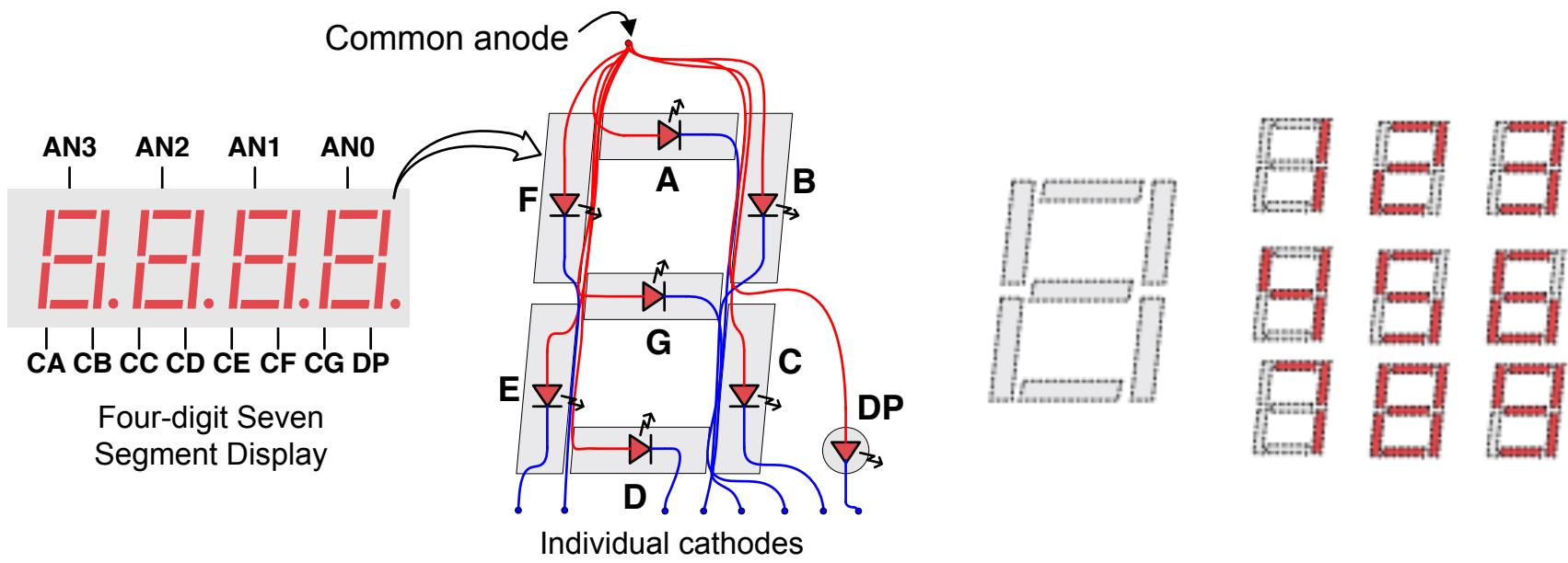
sel=0, b=1, a=0 ==> out=1

# Some Notes

- Sometimes, the database of the design will be corrupted, and any changes will not take effect or your board behaves weird.
  - Open a new project with fresh source files.
- Look into the ‘Errors’ or ‘Warnings’ windows to debug your design.
- If you finish your lab at dorm and want to bring it to the lab for demo
  - DO NOT copy the entire directory to the lab and use the same directory for demo
    - Just copying the .v and .xdc files to the lab is sufficient.
    - Use ‘New Project’ in the lab and open the existing source files to re-implement your design for demo

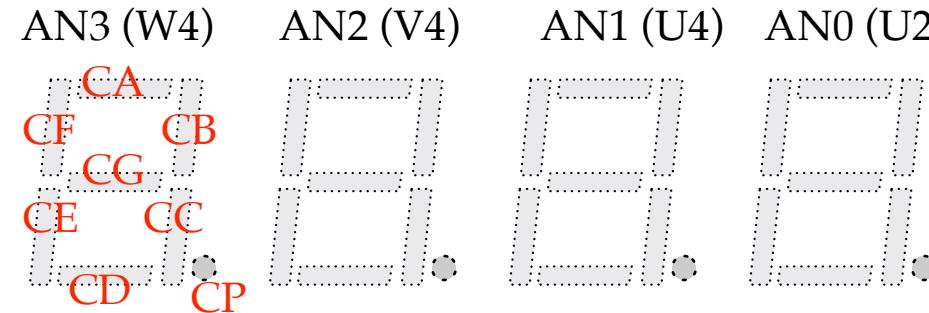
# 7-Segment Display (1 / 2)

- The anodes of segments forming each digit on all four displays are connected to the same FPGA pin (AN3, AN2, AN1, AN0) (common anode)



# 7-Segment Display (2/2)

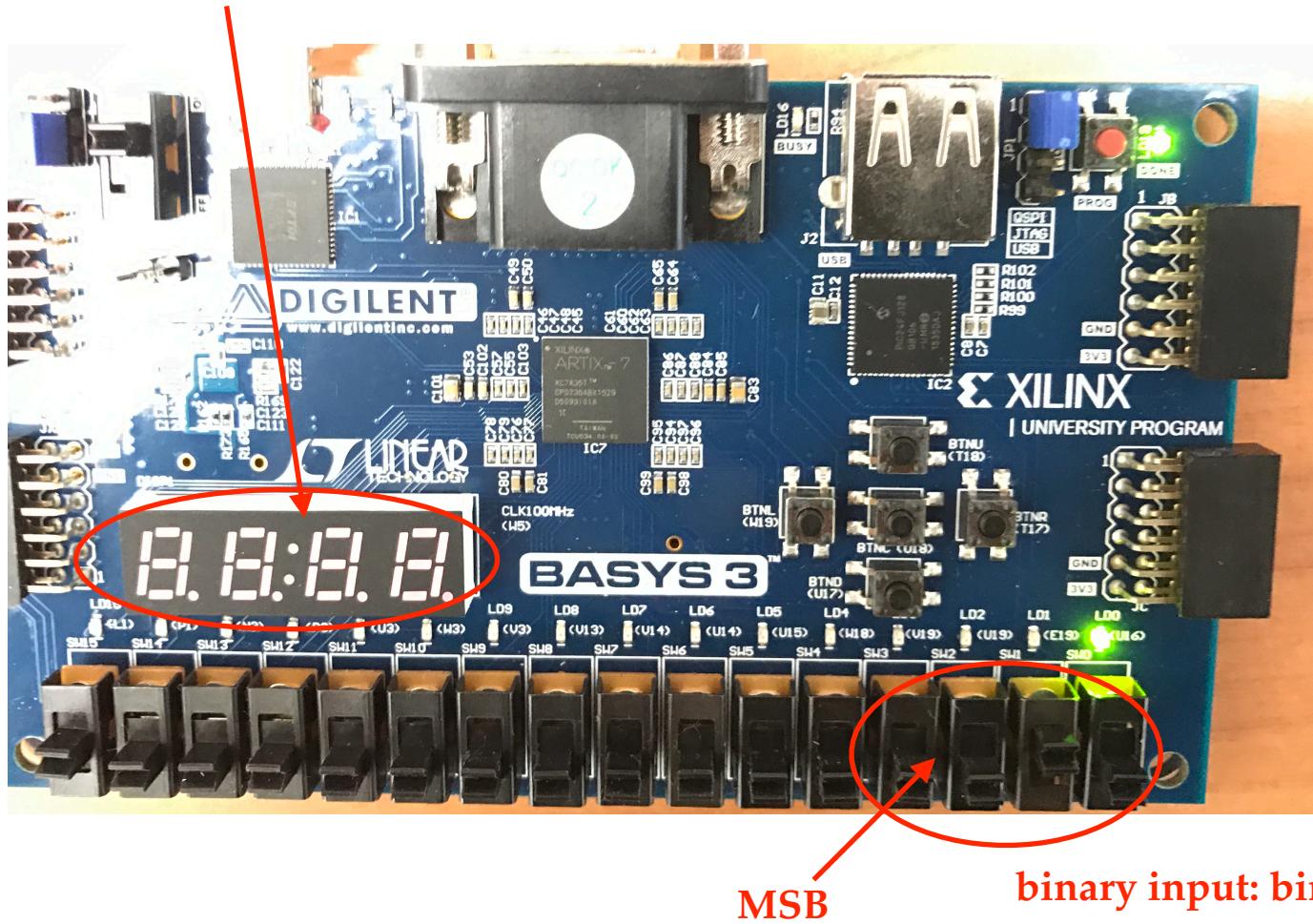
- 8 pins to control each 7-segment display
  - Including the point
- 4 pins (W4,V4,U4,U2) to choose which 7-seg to display
- Device is low activated



Symbol	CA	CB	CC	CD	CE	CF	CG	CP
FPGA Pin	W7	W6	U8	V8	U5	V5	U7	V7

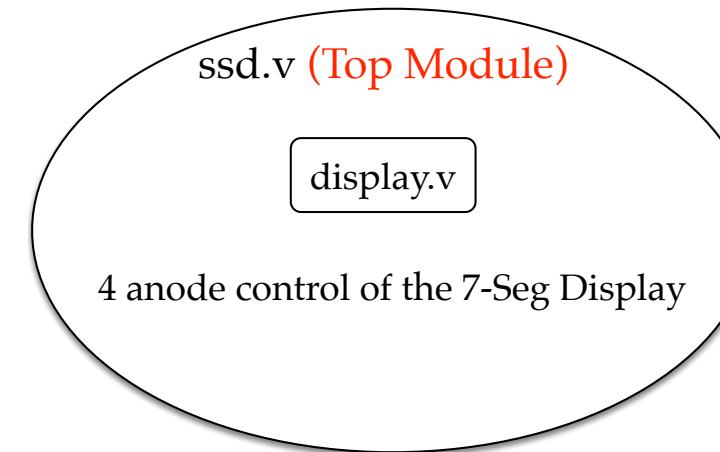
# Example

All 4 digits of 7-Seg Display show same number (BCD number output)

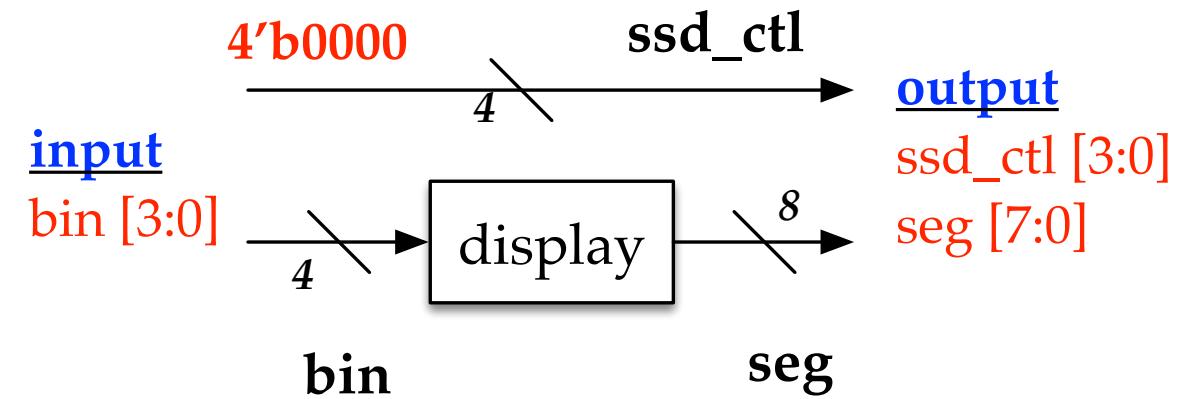


# Block Diagram for 7-Seg Display

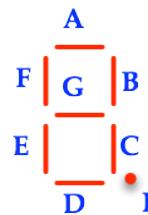
## System Hierarchy



## Block Diagram



# A BCD to Seven-Segment Display Decoder



```
// define segment codes
`define SS_0 8'b00000011
`define SS_1 8'b10011111
`define SS_2 8'b00100101
`define SS_3 8'b00001101
`define SS_4 8'b10011001
`define SS_5 8'b01001001
`define SS_6 8'b01000001
`define SS_7 8'b00011111
`define SS_8 8'b00000001
`define SS_9 8'b00001001
```

```
module display(segs, bin);
output [7:0] segs;
input [3:0] bin;
reg [7:0] segs;

always @*
  case (bin)
    4'd0: segs = `SS_0;
    4'd1: segs = `SS_1;
    4'd2: segs = `SS_2;
    4'd3: segs = `SS_3;
    4'd4: segs = `SS_4;
    4'd5: segs = `SS_5;
    4'd6: segs = `SS_6;
    4'd7: segs = `SS_7;
    4'd8: segs = `SS_8;
    4'd9: segs = `SS_9;
    default: segs = 8'b00000000;
  endcase
endmodule
```

# Top Module (ssd.v)

```
module ssd(seg, bin, ssd_ctl);
    output [3:0] ssd_ctl;
    output [7:0] seg;
    input [3:0] bin;

    display U0(.seg(.seg(seg), .bin(bin)));

    assign ssd_ctl = 4'b0000;

endmodule
```

# ssd.xdc (1 / 2)

```
# Four anode control signals
set_property PACKAGE_PIN W4 [get_ports {ssd_ctl[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {ssd_ctl[3]}]
set_property PACKAGE_PIN V4 [get_ports {ssd_ctl[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {ssd_ctl[2]}]
set_property PACKAGE_PIN U4 [get_ports {ssd_ctl[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {ssd_ctl[1]}]
set_property PACKAGE_PIN U2 [get_ports {ssd_ctl[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {ssd_ctl[0]}]

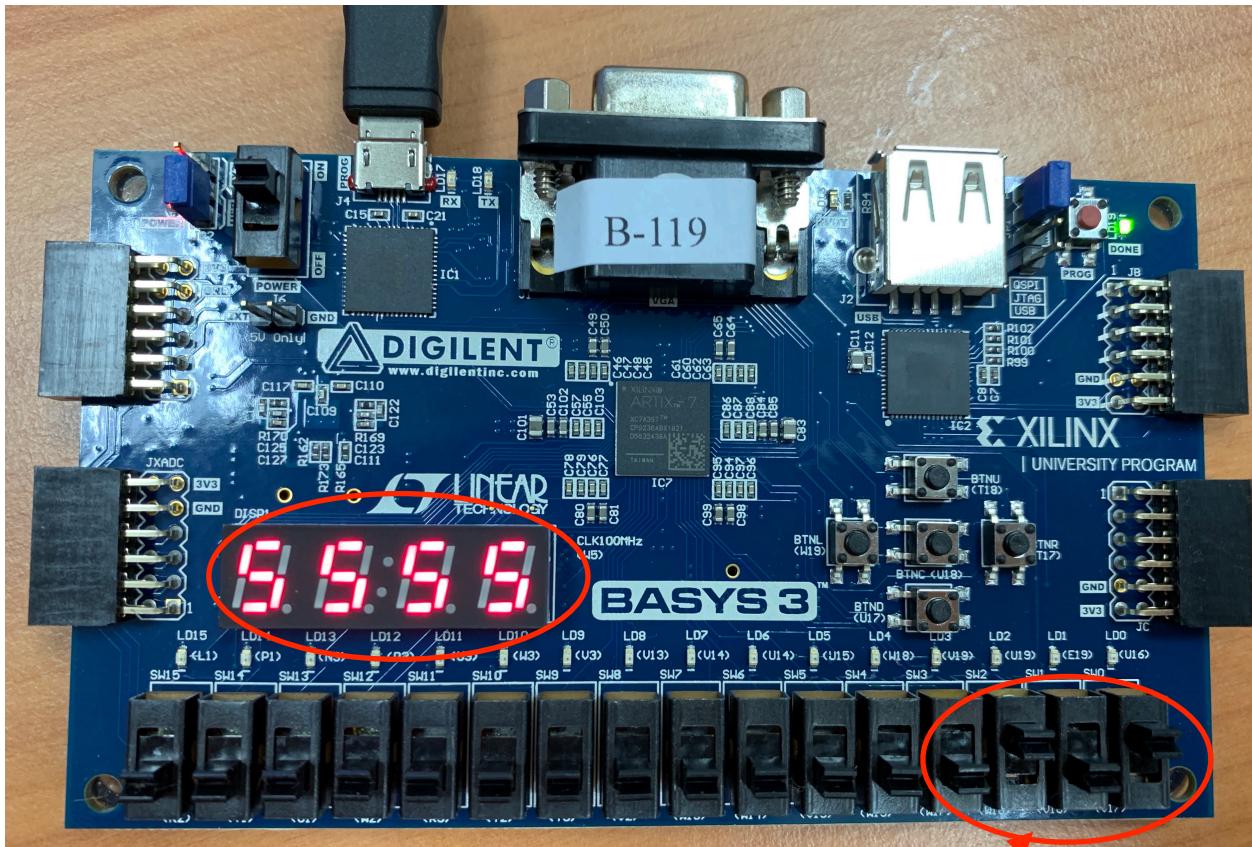
#8 common segment controls
set_property PACKAGE_PIN W7 [get_ports {seg[7]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[7]}]
set_property PACKAGE_PIN W6 [get_ports {seg[6]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[6]}]
set_property PACKAGE_PIN U8 [get_ports {seg[5]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[5]}]
set_property PACKAGE_PIN V8 [get_ports {seg[4]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[4]}]
```

# ssd.xdc (2/2)

```
set_property PACKAGE_PIN U5 [get_ports {seg[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[3]}]
set_property PACKAGE_PIN V5 [get_ports {seg[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[2]}]
set_property PACKAGE_PIN U7 [get_ports {seg[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[1]}]
set_property PACKAGE_PIN V7 [get_ports {seg[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seg[0]}]

#4-bit binary input
set_property PACKAGE_PIN W17 [get_ports {bin[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {bin[3]}]
set_property PACKAGE_PIN W16 [get_ports {bin[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {bin[2]}]
set_property PACKAGE_PIN V16 [get_ports {bin[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {bin[1]}]
set_property PACKAGE_PIN V17 [get_ports {bin[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {bin[0]}]
```

# Example



binary input: 0101