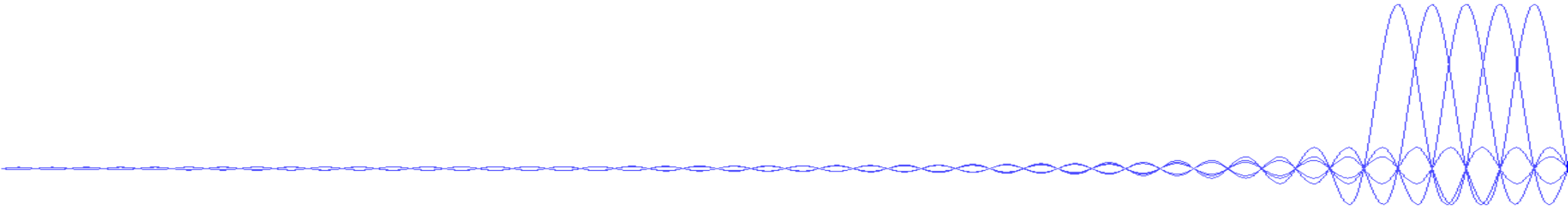
Raspberry Pi and ZERO7

From Setup to Design and Operation



Copyrights 2026 DSP Laboratory. All Rights Reserved.

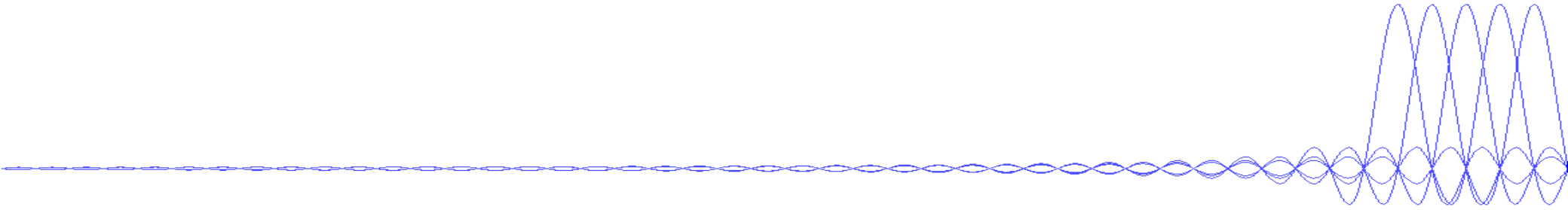
Development Workflow

- Raspberry Pi Side

- ZERO7 Side

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Raspberry Pi Development

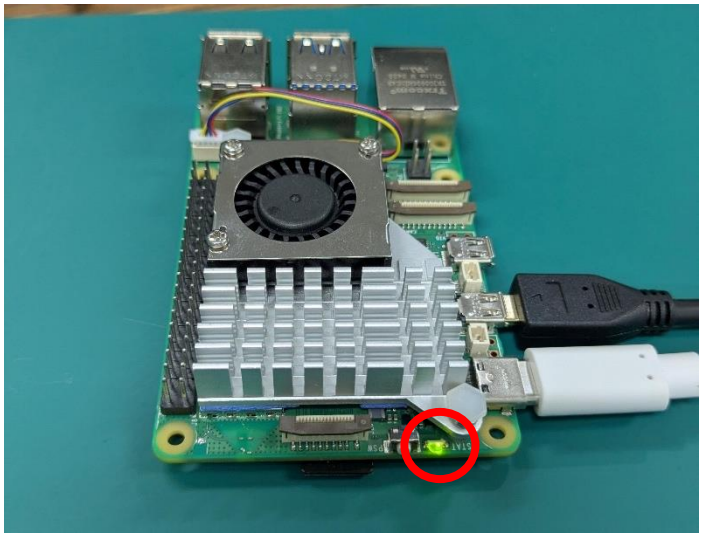


Copyrights 2026 DSP Laboratory. All Rights Reserved.

Raspberry Pi Development Setup 1

- Install an OS for Raspberry Pi
  - Refer to the following guide:
  - https://www.raspberrypi.com/documentation/computers/getting-started.html

- Connect Raspberry Pi to a monitor
- Connect Raspberry Pi to a power source
  - It starts automatically when powered on, and the LED lights up green as shown below.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Raspberry Pi Development Setup 2

- After starting Raspberry Pi, its desktop appears on the monitor.



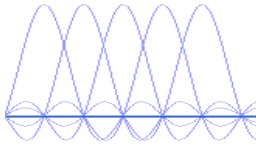
Raspberry Pi desktop

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Raspberry Pi Development Setup 3

- Connect Input/Output Devices
  - Connect a mouse and keyboard via USB or Bluetooth to develop on Raspberry Pi.
- Internet Connection
  - Connect to the internet using Wi-Fi or an Ethernet cable.
  - For more details, see:
  - https://www.raspberrypi.com/documentation/computers/getting-started.html#wi-fi
- Other Options
  - You can also develop over SSH from a Windows PC. See the Appendix for details.

Copyrights 2026 DSP Laboratory. All Rights Reserved.



Running Programs on Raspberry Pi

- Prepare the Python Program
  - The program used here is “raspi-zer07_button.py”.
  - It is provided in the same folder as this manual.
  - Copy it to any folder on Raspberry Pi using a USB flash drive or another method.

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Running Programs on Raspberry Pi 2

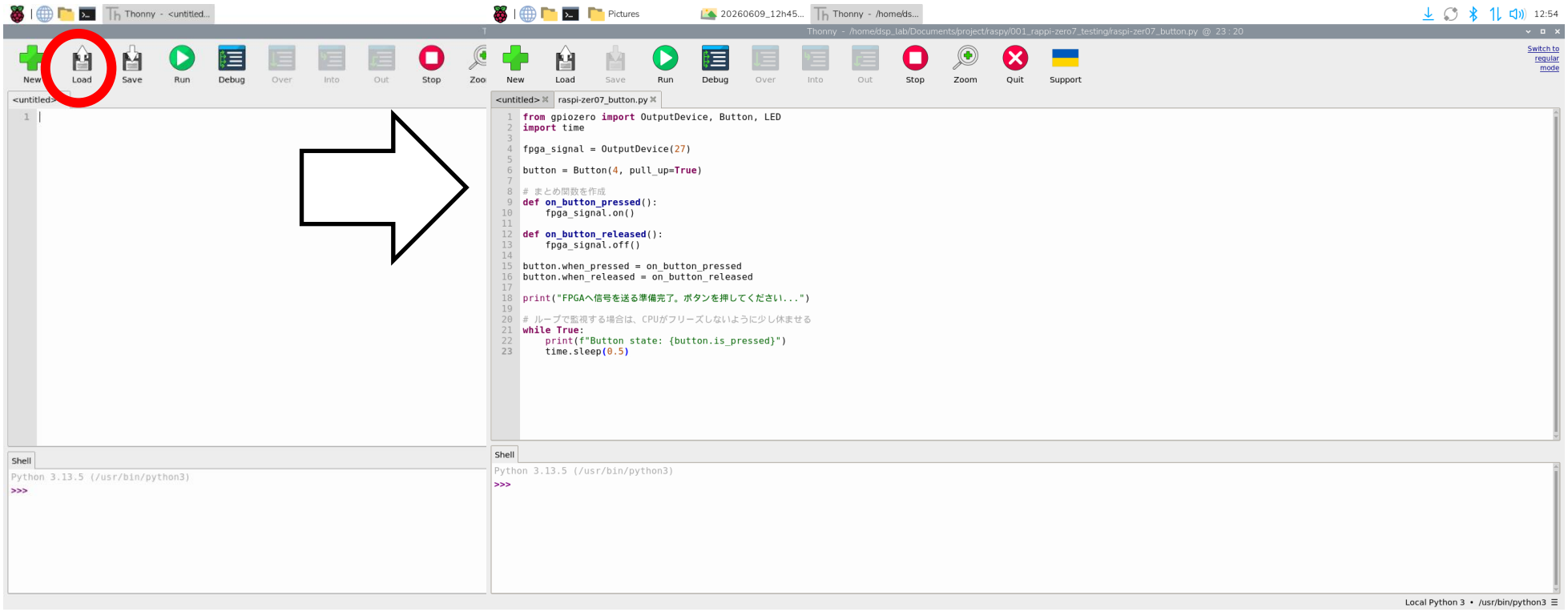
- Prepare Software for Running Python
  - Use Thonny, which is installed on Raspberry Pi by default.
  - Click the Raspberry Pi icon (top left) → Programming → Thonny.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Running Programs on Raspberry Pi 3

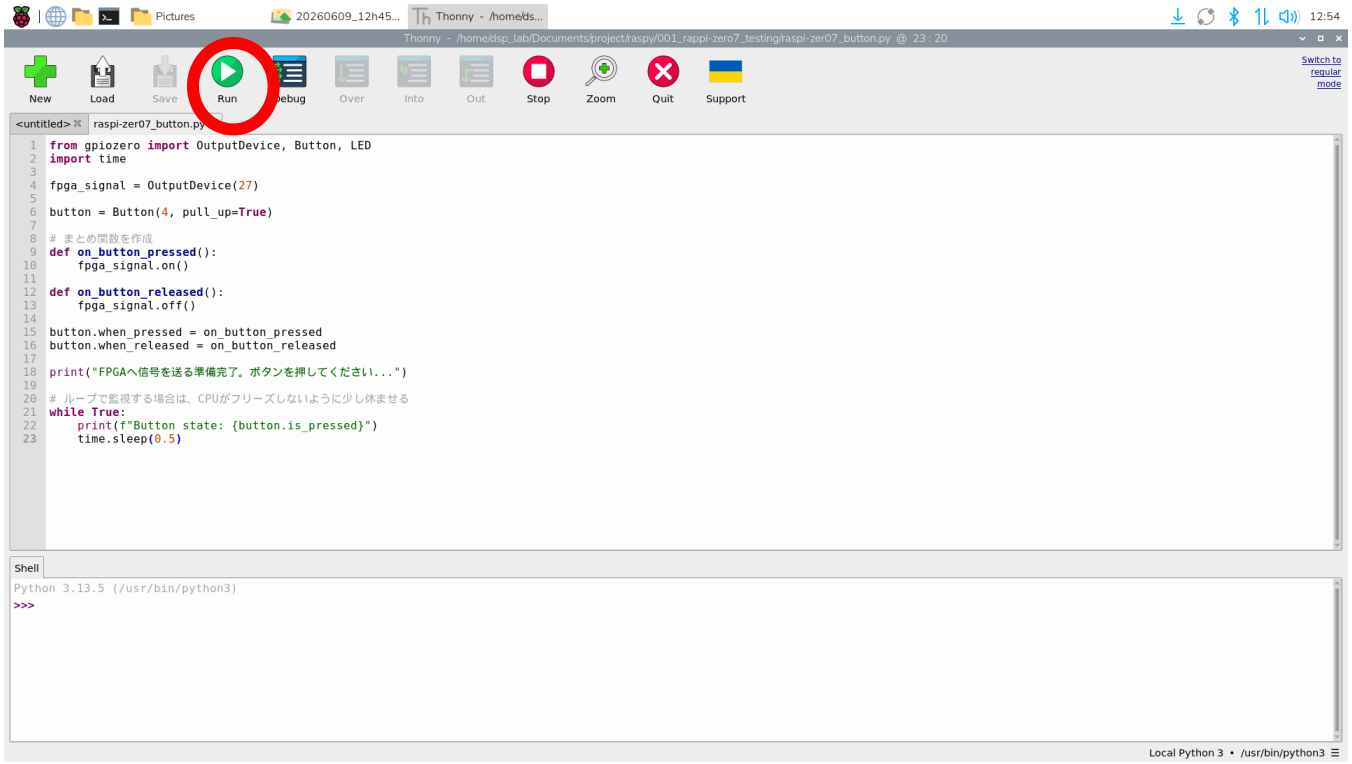
- Load the Python Program
  - Select “Load” in the Thonny window.
  - Select the downloaded program file.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Running Programs on Raspberry Pi 4

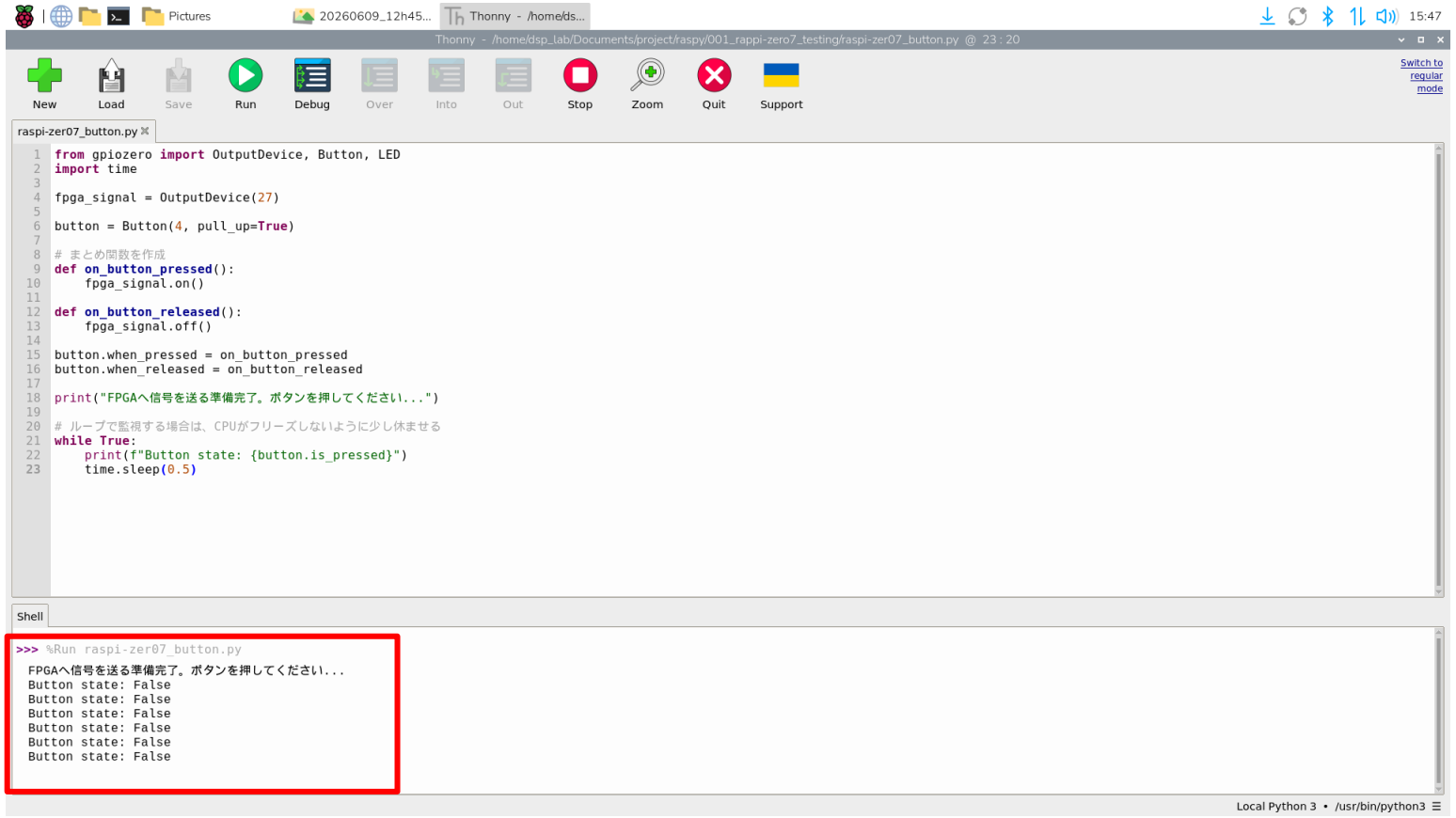
- Run the Python Program
  - Open the Python program you want to run.
  - Click “Run” to execute the program.



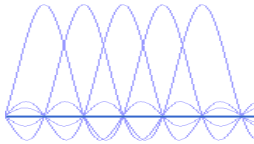
Copyrights 2026 DSP Laboratory. All Rights Reserved.

Verify Program Execution

- Running this program produces the Shell output shown below.
  (The Shell display will not change yet because the circuit has not been built.)



Copyrights 2026 DSP Laboratory. All Rights Reserved.

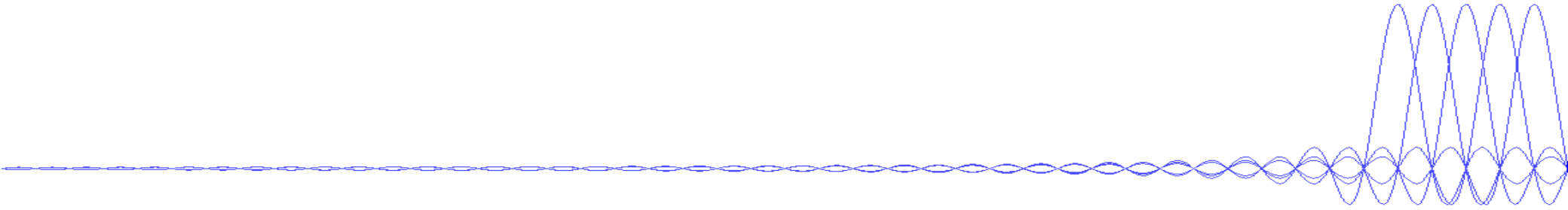


Raspberry Pi Development

- Raspberry Pi is now ready to run the program.

Copyrights 2026 DSP Laboratory. All Rights Reserved.

ZERO7 Development



Copyrights 2026 DSP Laboratory. All Rights Reserved.

ZERO7 Specifications / Development Environment

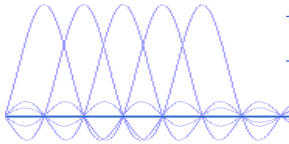
- Refer to the ZERO7 User Manual
- ZERO7 FPGA
  - Series
    - Spartan 7
  - Part Number
    - xc7s25csga225-1 ⟵ Select this device in Vivado
- Environment
  - Windows 11
  - Vivado 2025.2

Copyrights 2026 DSP Laboratory. All Rights Reserved.

ZERO7 Development Workflow

- Install Vivado
- Prepare the Required Program Files
- Create a Vivado Project
  - Add Program Files (Add Sources)
  - Run a Testbench ← Skipped This Time
  - Create an ILA
  - Generate the Bitstream
  - Prepare ZERO7
  - Connect ZERO7 to a Windows PC (JTAG)
  - Upload the .bit File to ZERO7

Copyrights 2026 DSP Laboratory. All Rights Reserved.



Install Vivado

- Vivado installation is explained in the following slides.

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Installing Vivado 1

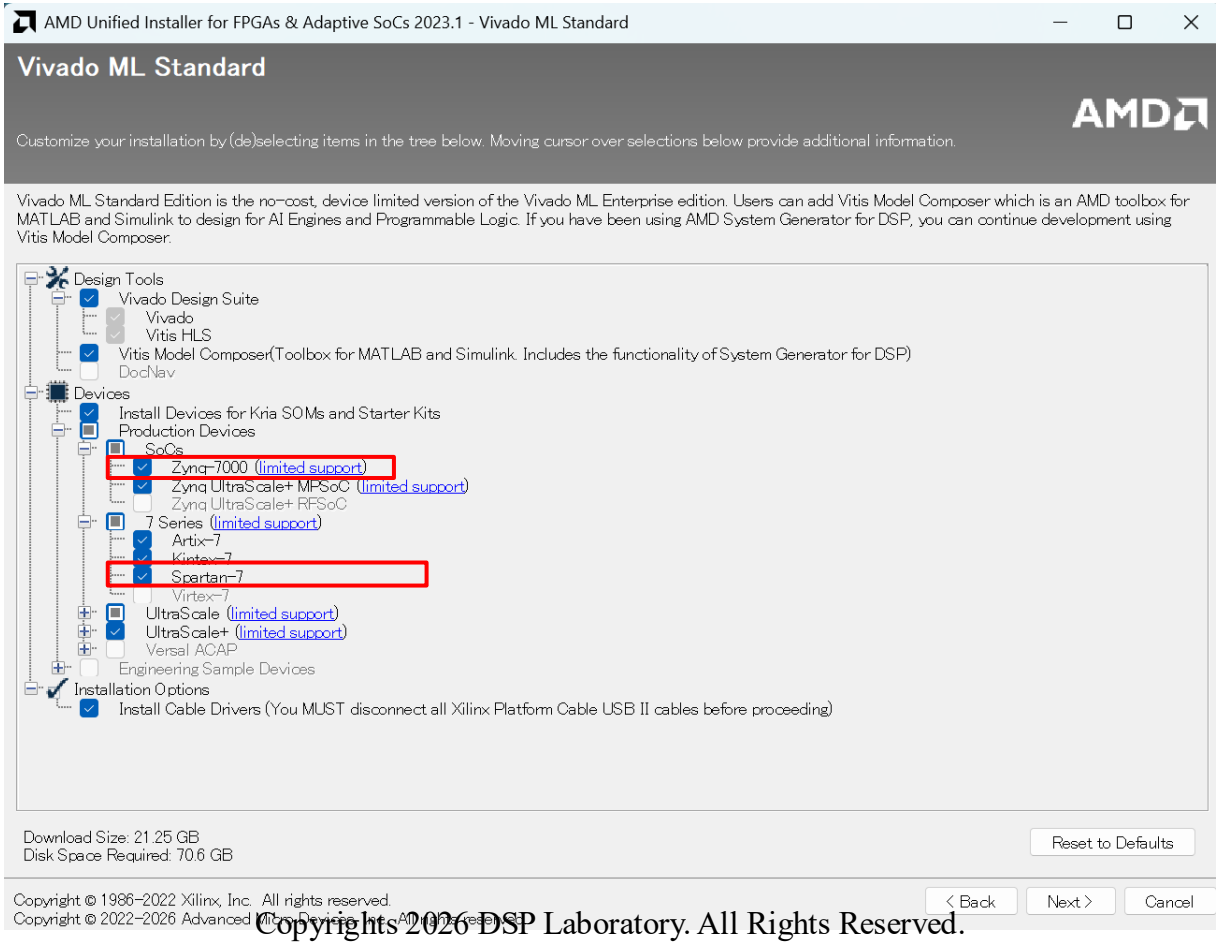
- Refer to the following installation guide:
  - How to Install Vivado [Xilinx FPGA Setup] on Windows – kotoha sensorium

- Follow the installation steps at the linked website.
  The following slides cover only steps that differ from the website instructions.

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Installing Vivado 2

- Target Device Families (Minimum)
  - 7Series → Spartan-7
  - SoCs → Zynq-7000



Copyrights 2026 DSP Laboratory. All Rights Reserved.

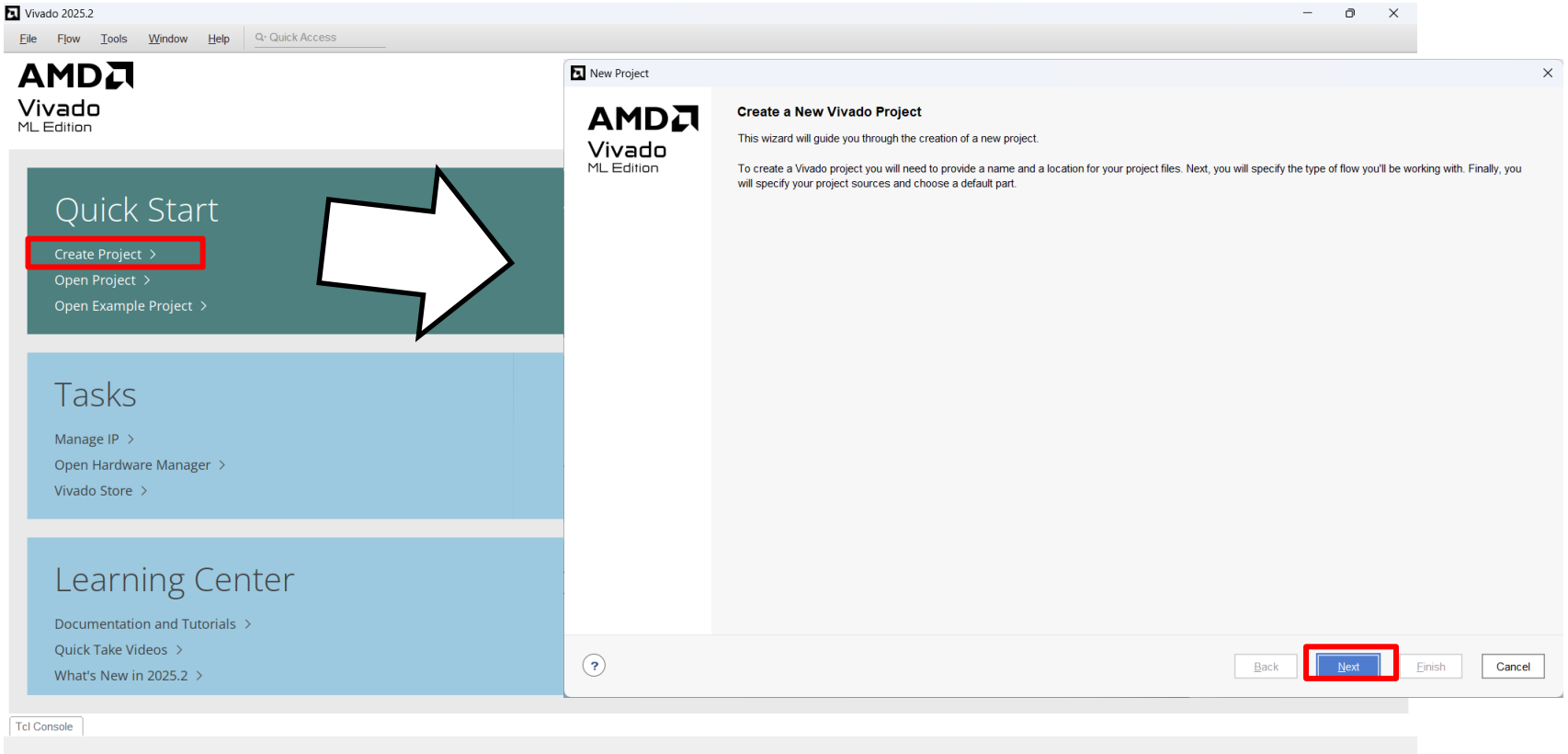
Prepare the Verilog Program

- Prepare the Required Program Files
  - Verilog file: “pi_button_receiver.v”
  - Constraints file: “Zero7.xdc”

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Creating a Vivado Project 1

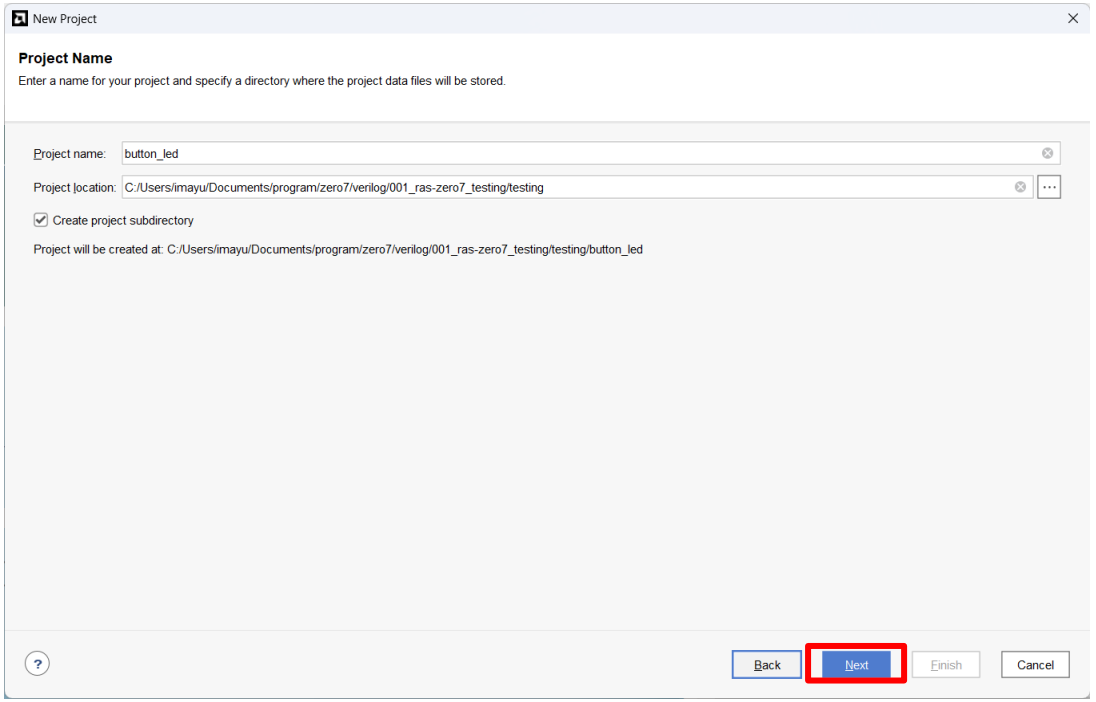
- Launch Vivado and create a new project.
  - Select “Create Project”.
  - Select “Next”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Creating a Vivado Project 2

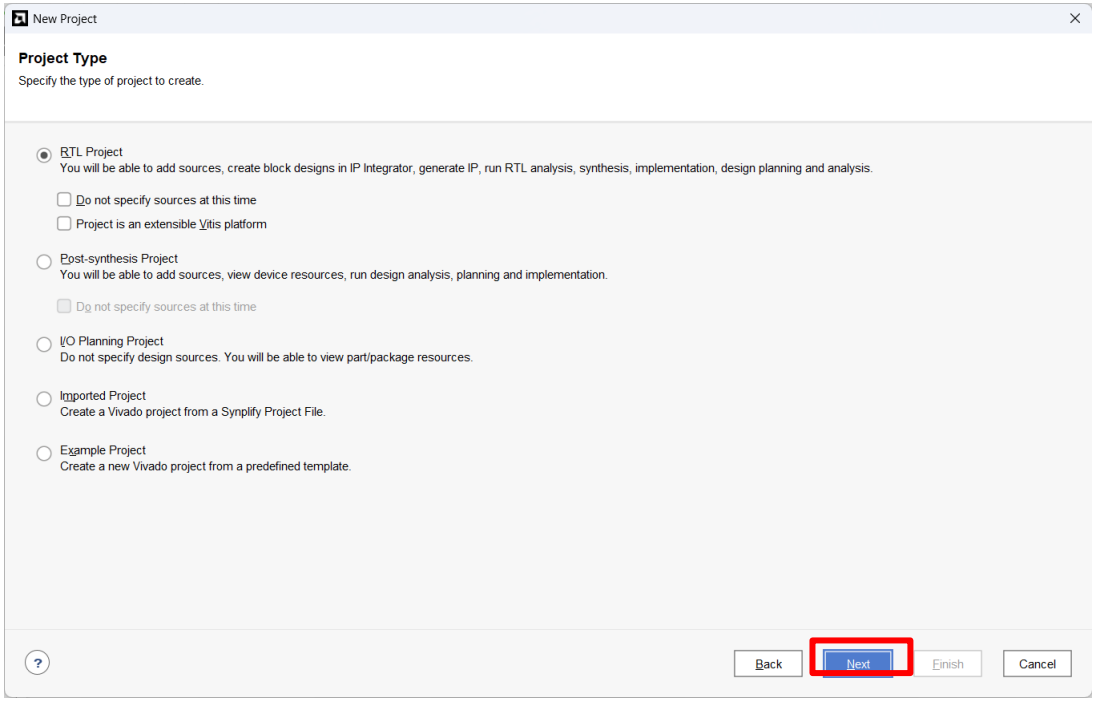
- Choose a Project Folder and Name
  - Choose any name, but keep it reasonably short.
  - Select “Next”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Creating a Vivado Project 3

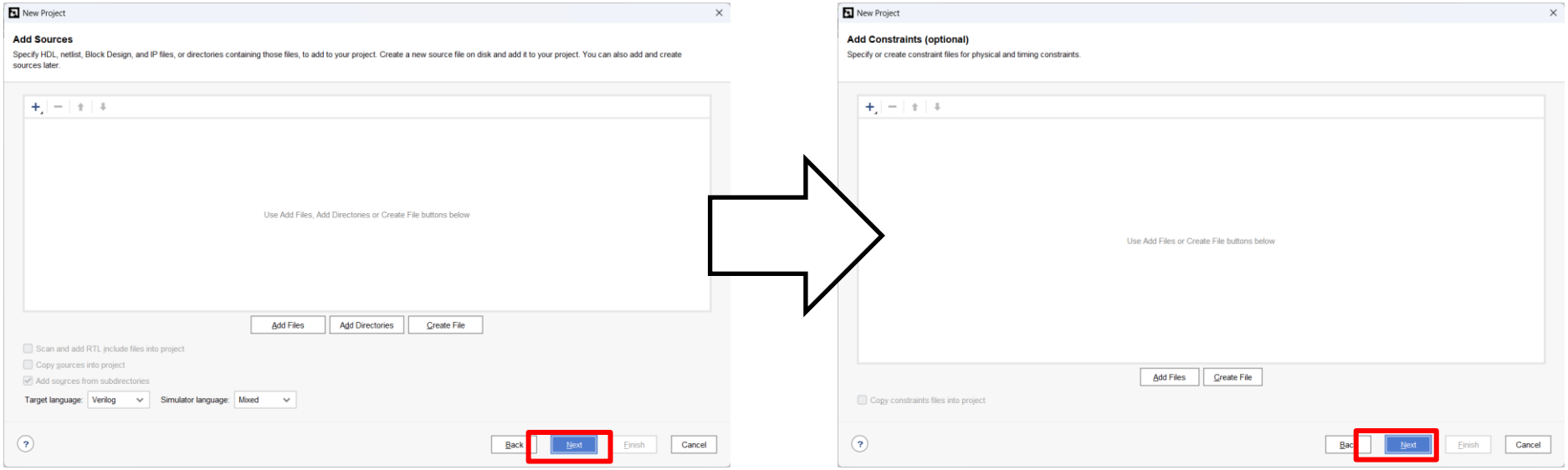
- Configure Project Settings
  - Generally, click “Next” on each screen.
  - First, select “Project Type”.
  - Keep the default “RTL Project” option.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Creating a Vivado Project 4

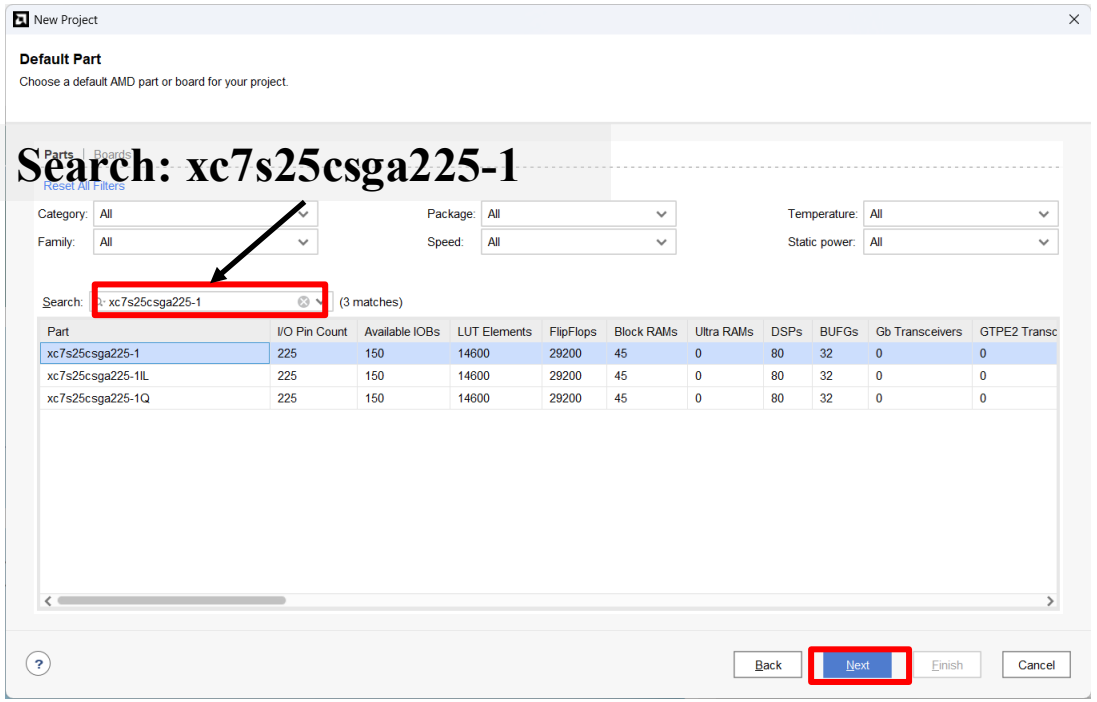
- Configure Project Settings
  - On “Add Sources”, click “Next”.
    - You may select the Verilog file here.
    - In this tutorial, we will add it later.
  - On “Add Constraints”, click “Next”.
    - You may select the XDC file here.
    - In this tutorial, we will add it later.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Creating a Vivado Project 5

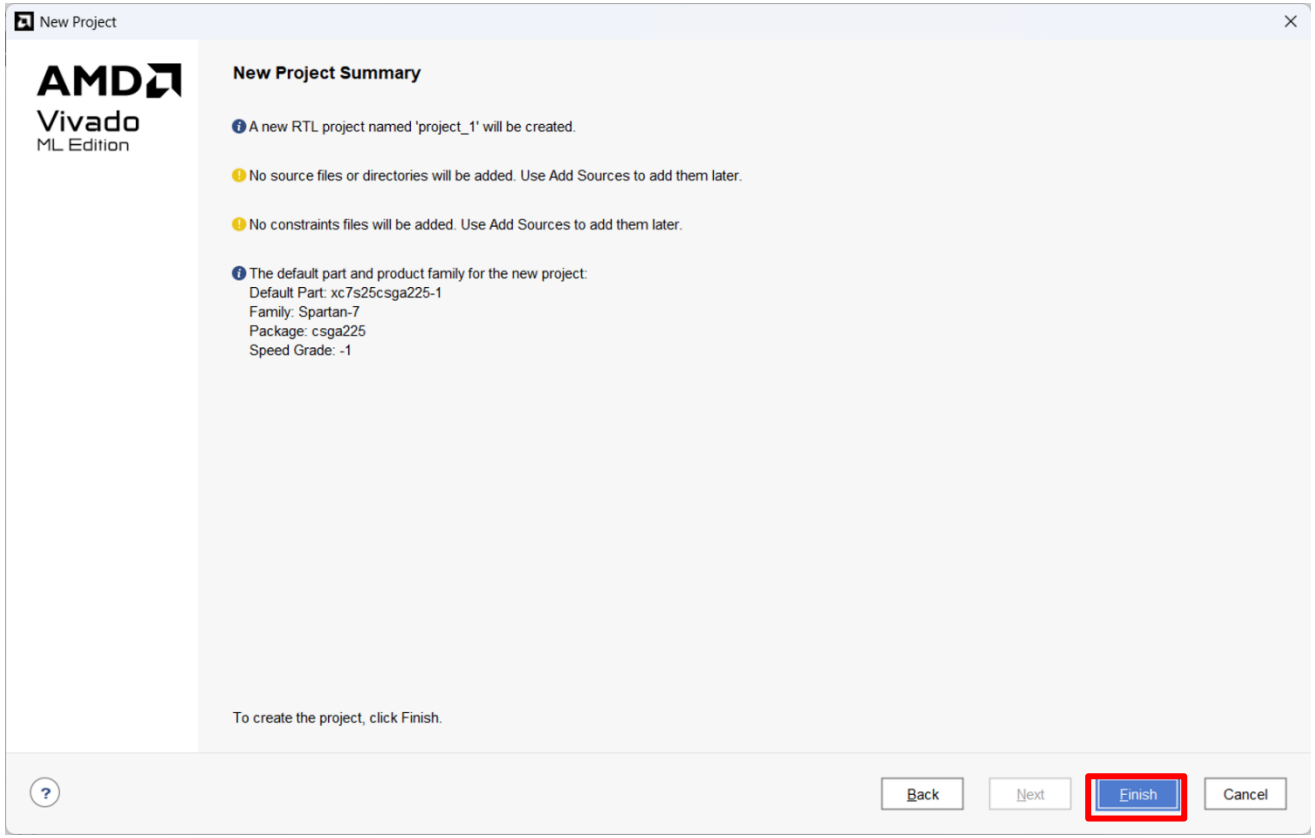
- Configure Project Settings
  - Select the Target FPGA Device
  - For ZERO7, search for and select “xc7s25csga225-1”.
  - When the selected device is highlighted in blue, click “Next”.



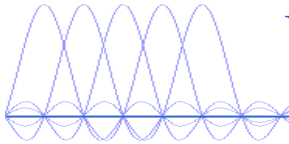
Copyrights 2026 DSP Laboratory. All Rights Reserved.

Creating a Vivado Project 6

- Review the Summary
  - The summary should look like the example below.
  - Select “Finish”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.



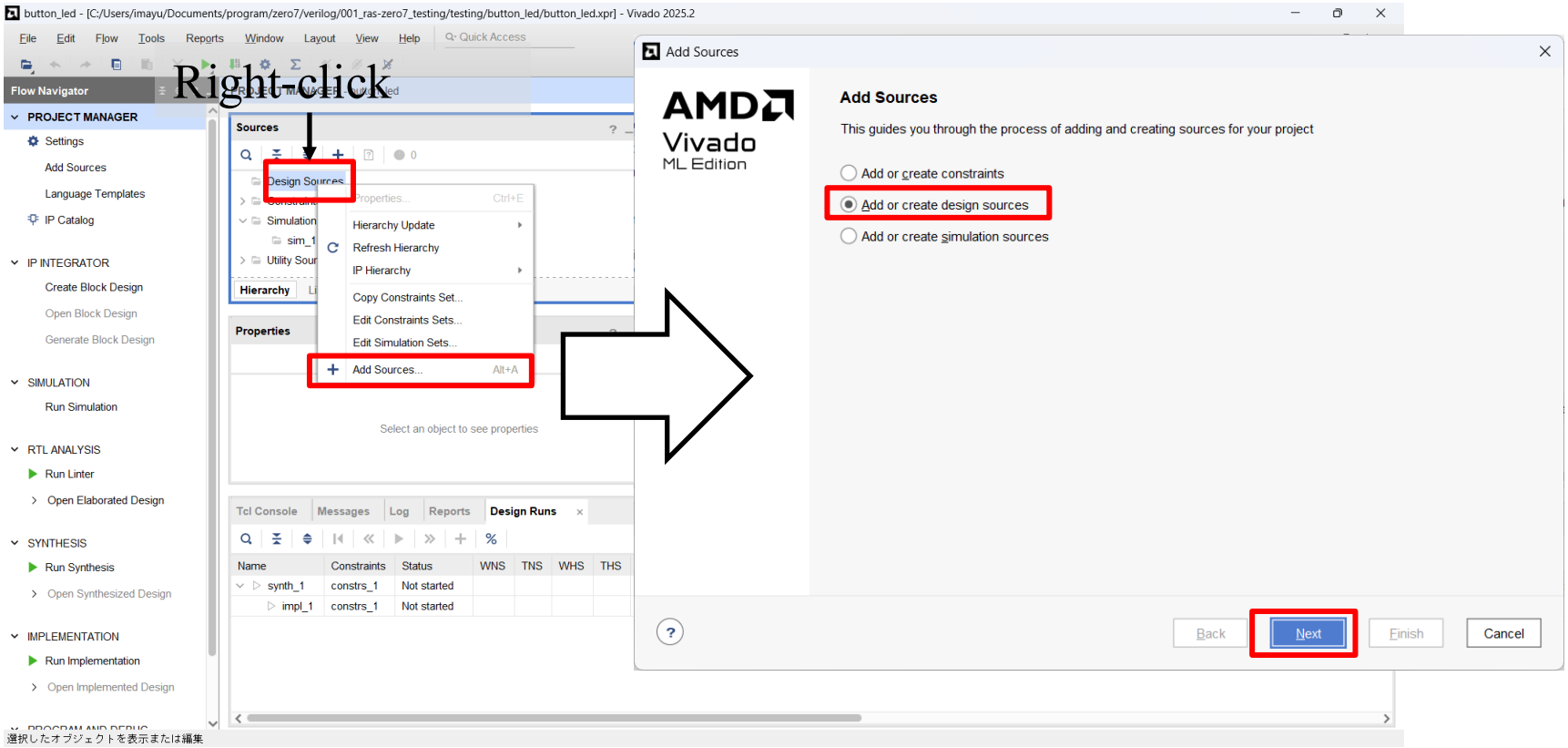
Vivado Project | Adding Files 1

- Add the Two Required Files
  - The Verilog file and the constraints file.

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Adding Files 2

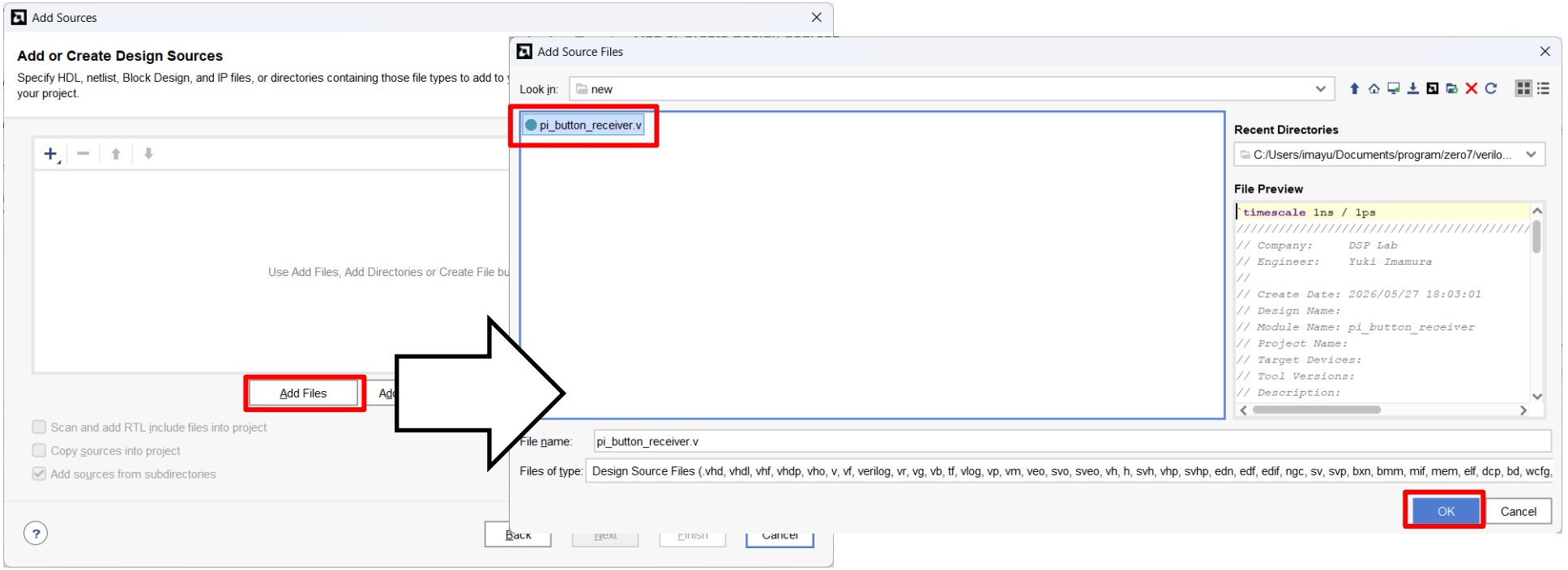
- Add the Verilog File
  - Right-click “Design Sources”.
  - Select “Add Sources”.
  - In the new window, select “Add or create design sources” and click “Next”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Adding Files 3

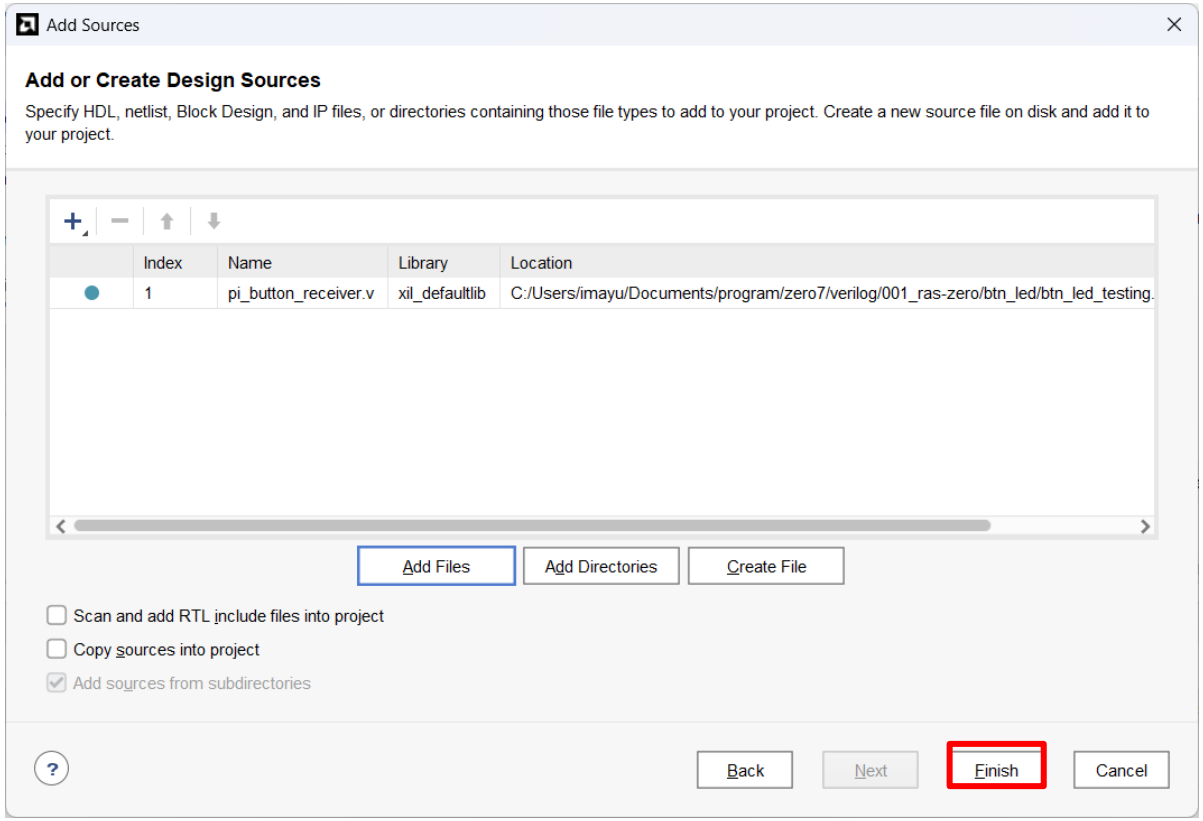
- Add the Verilog File
  - Select “Add Files”.
  - In the new window, select “pi_button_receiver.v” and click “OK”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Adding Files 4

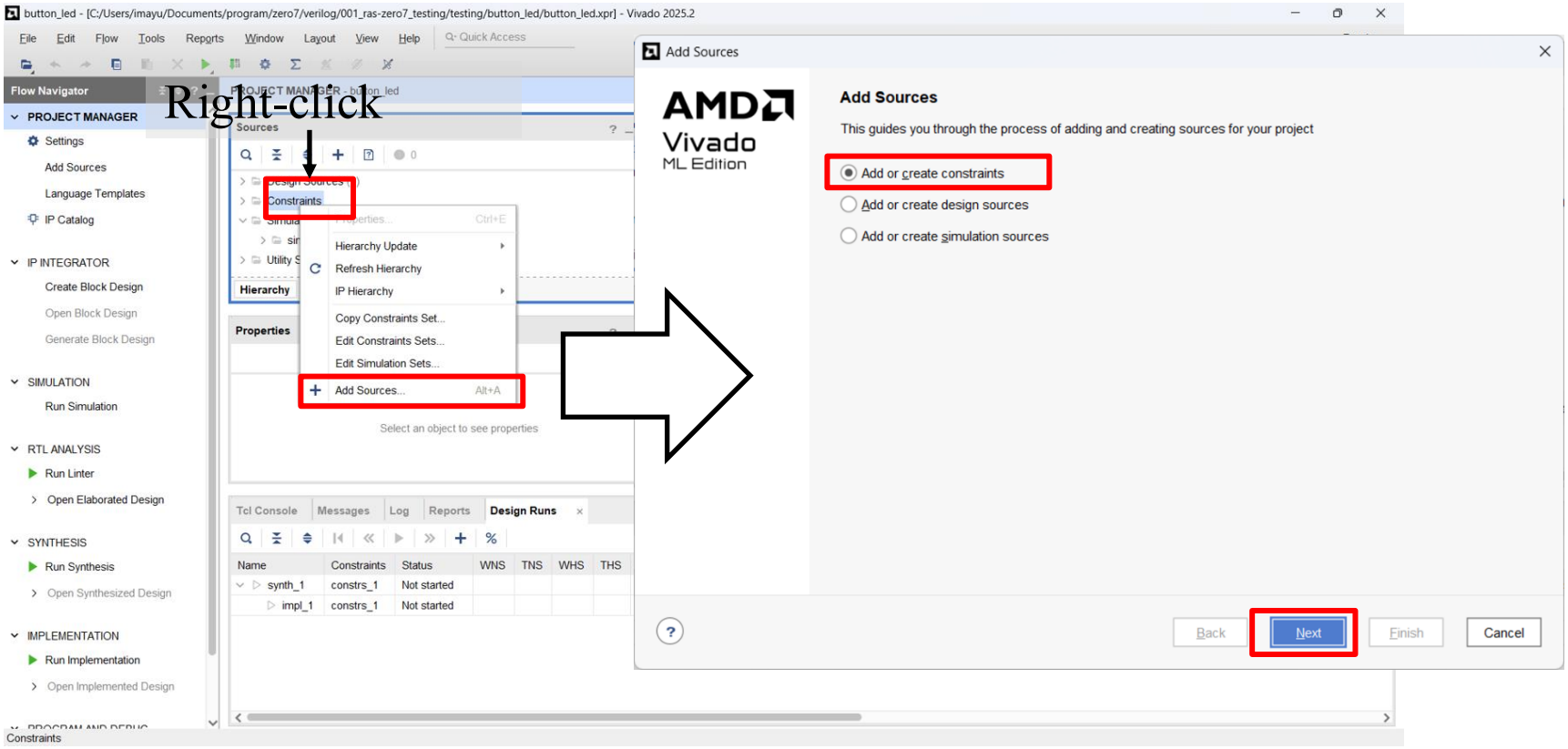
- Add the Verilog File
  - Confirm that the file appears, then click “Finish”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Adding Files 5

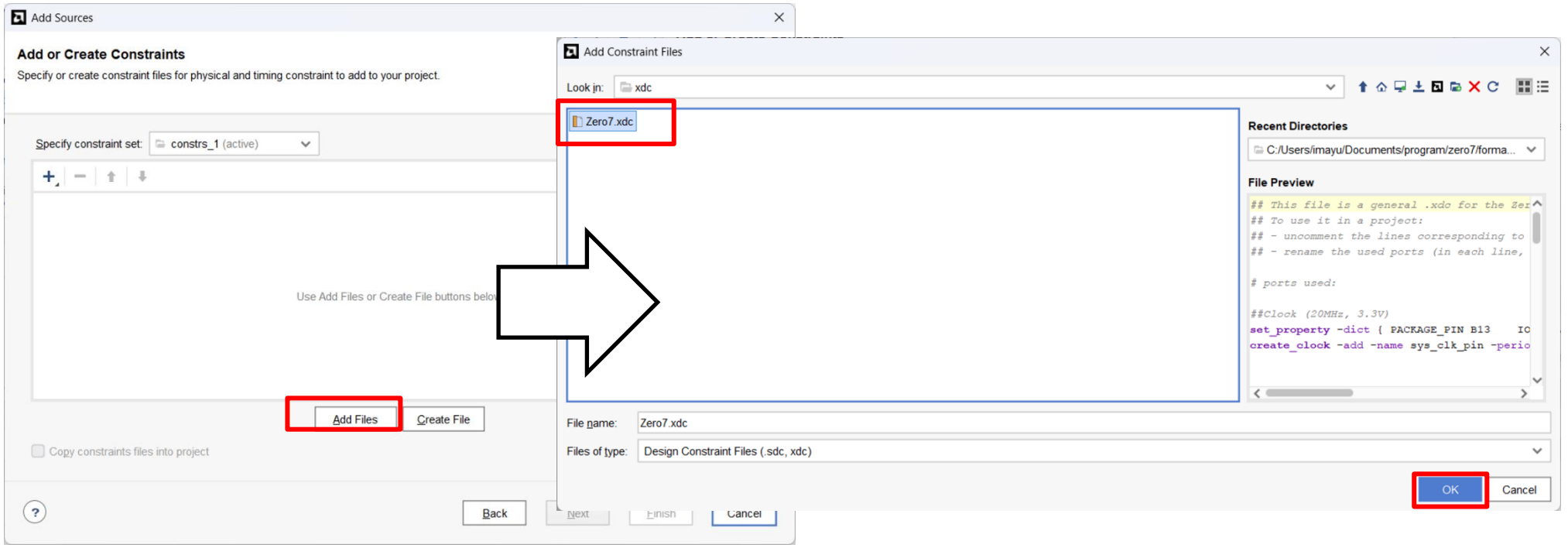
- Add the Constraints File
  - Right-click “Constraints”.
  - Select “Add Sources”.
  - In the new window, select “Add or create constraints” and click “Next”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Adding Files 6

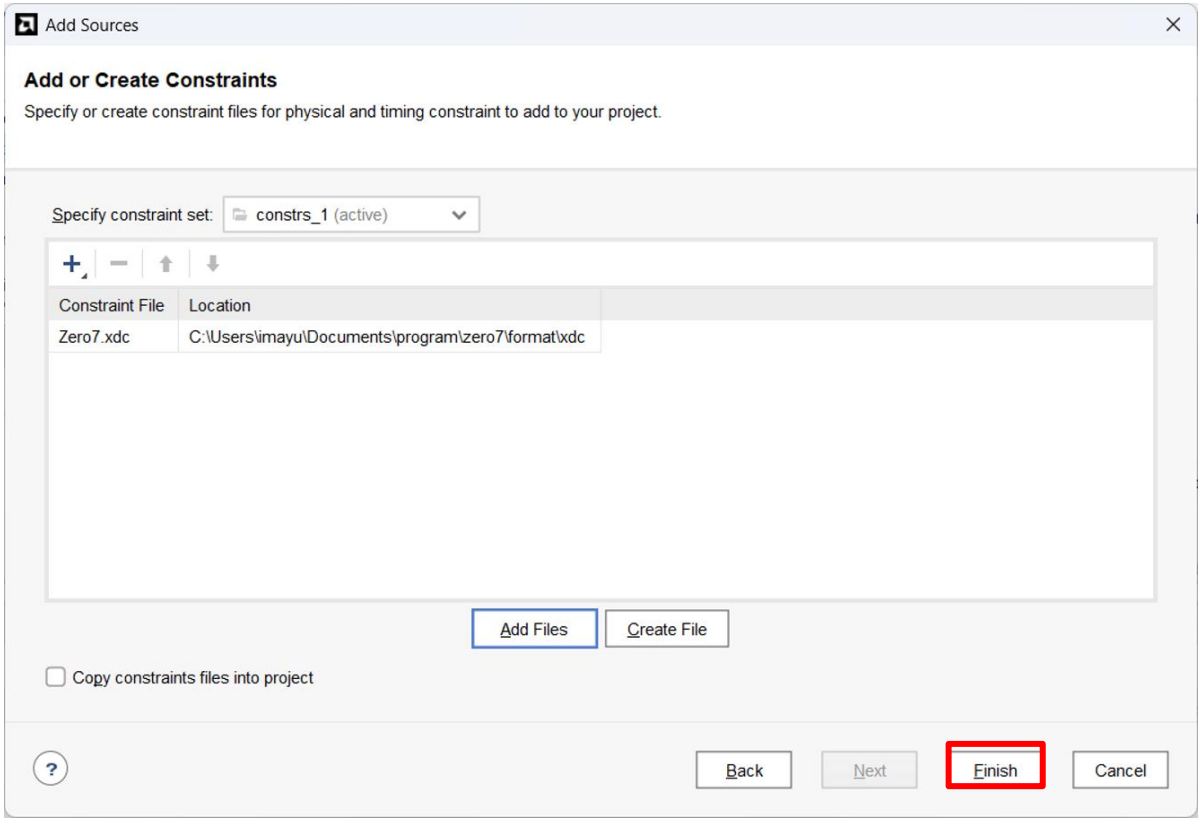
- Add the Constraints File
  - Select “Add Files”.
  - In the new window, select “Zero7.xdc” and click “OK”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Adding Files 7

- Add the Constraints File
  - Confirm that the file appears, then click “Finish”.

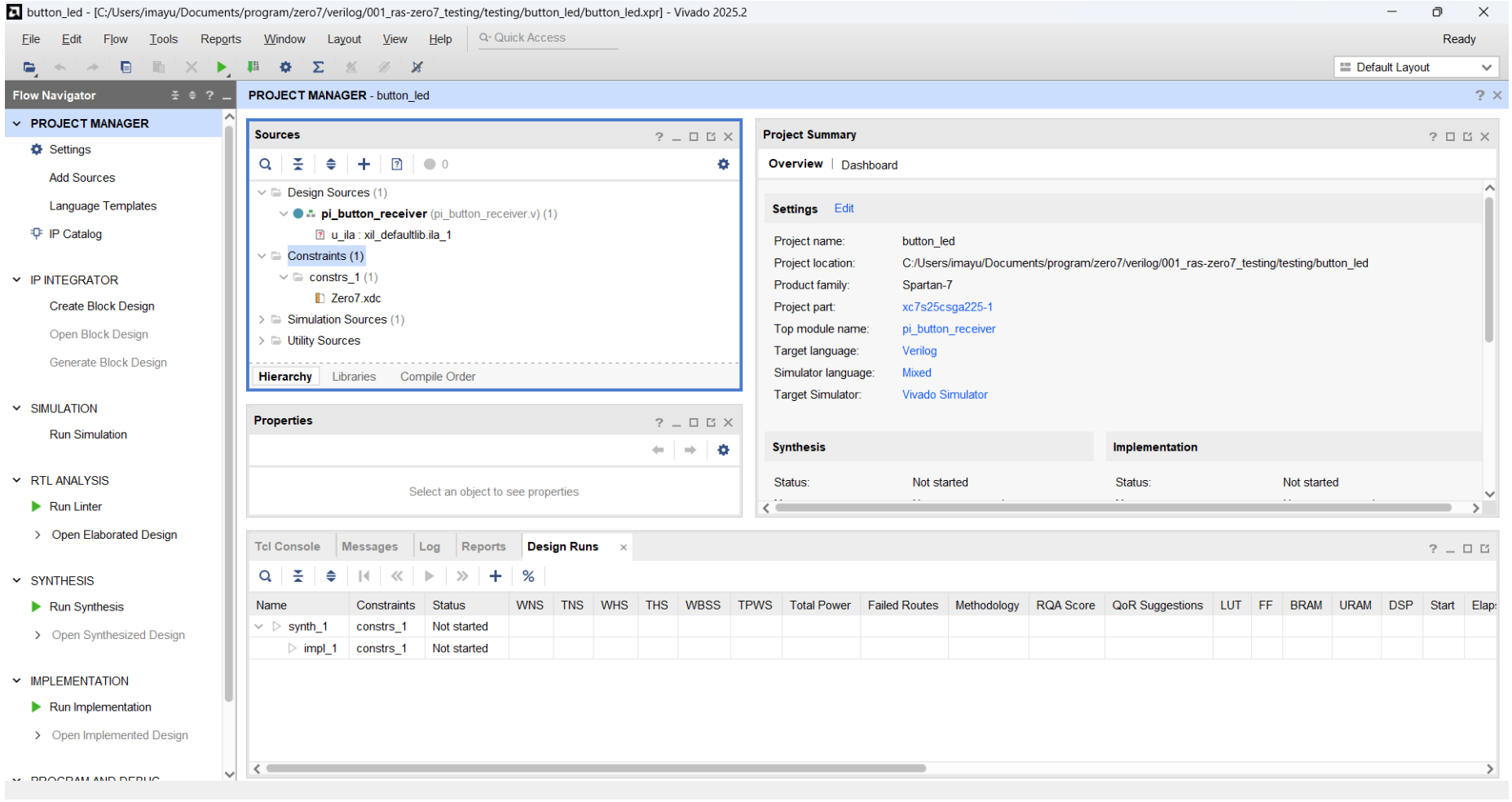


Copyrights 2026 DSP Laboratory. All Rights Reserved.

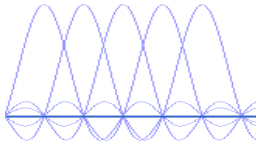
Vivado Project | Adding Files 8

Final File List

- Verify that the files appear as shown below.



Copyrights 2026 DSP Laboratory. All Rights Reserved.



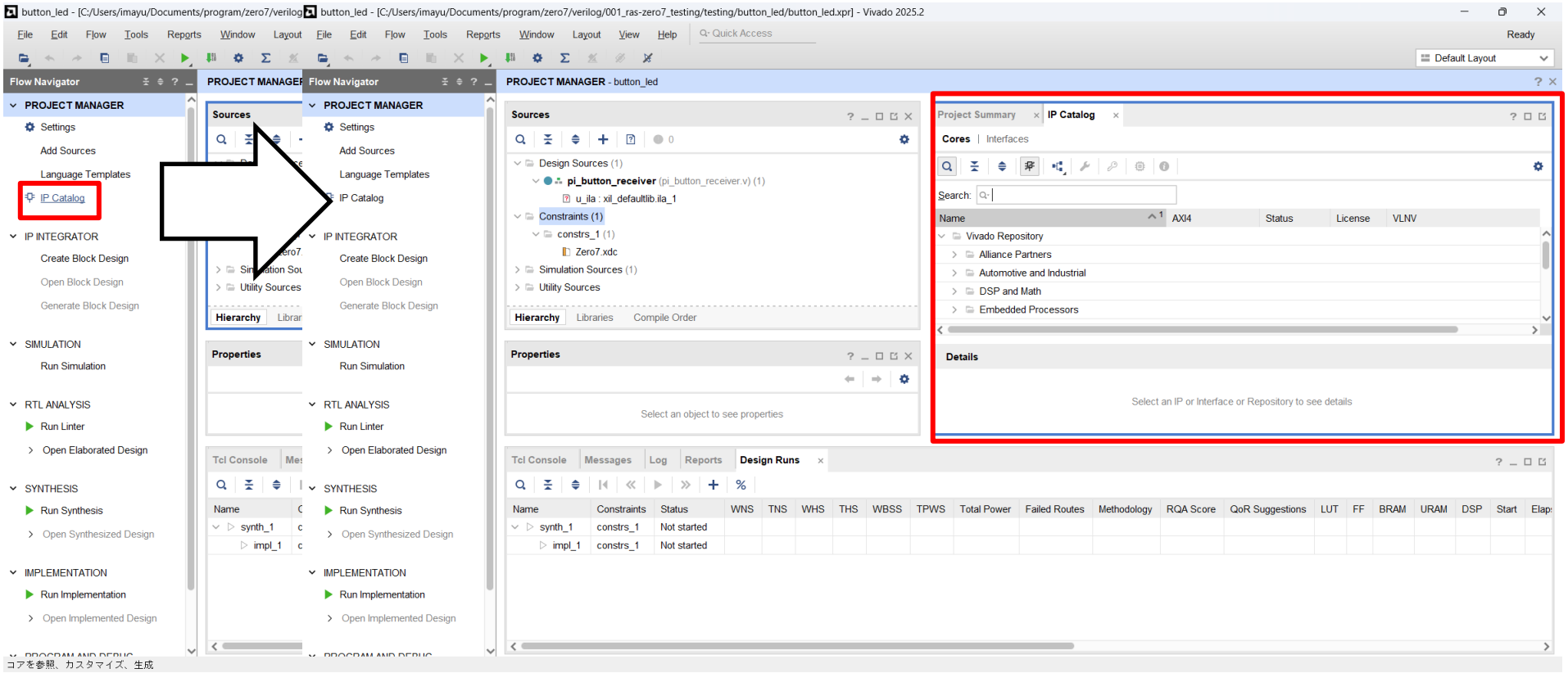
Vivado Project | Creating the ILA 1

- Add an ILA (Integrated Logic Analyzer)
  - Add an ILA to verify that the Spartan-7 FPGA on ZERO7 is operating correctly.

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Creating the ILA 2

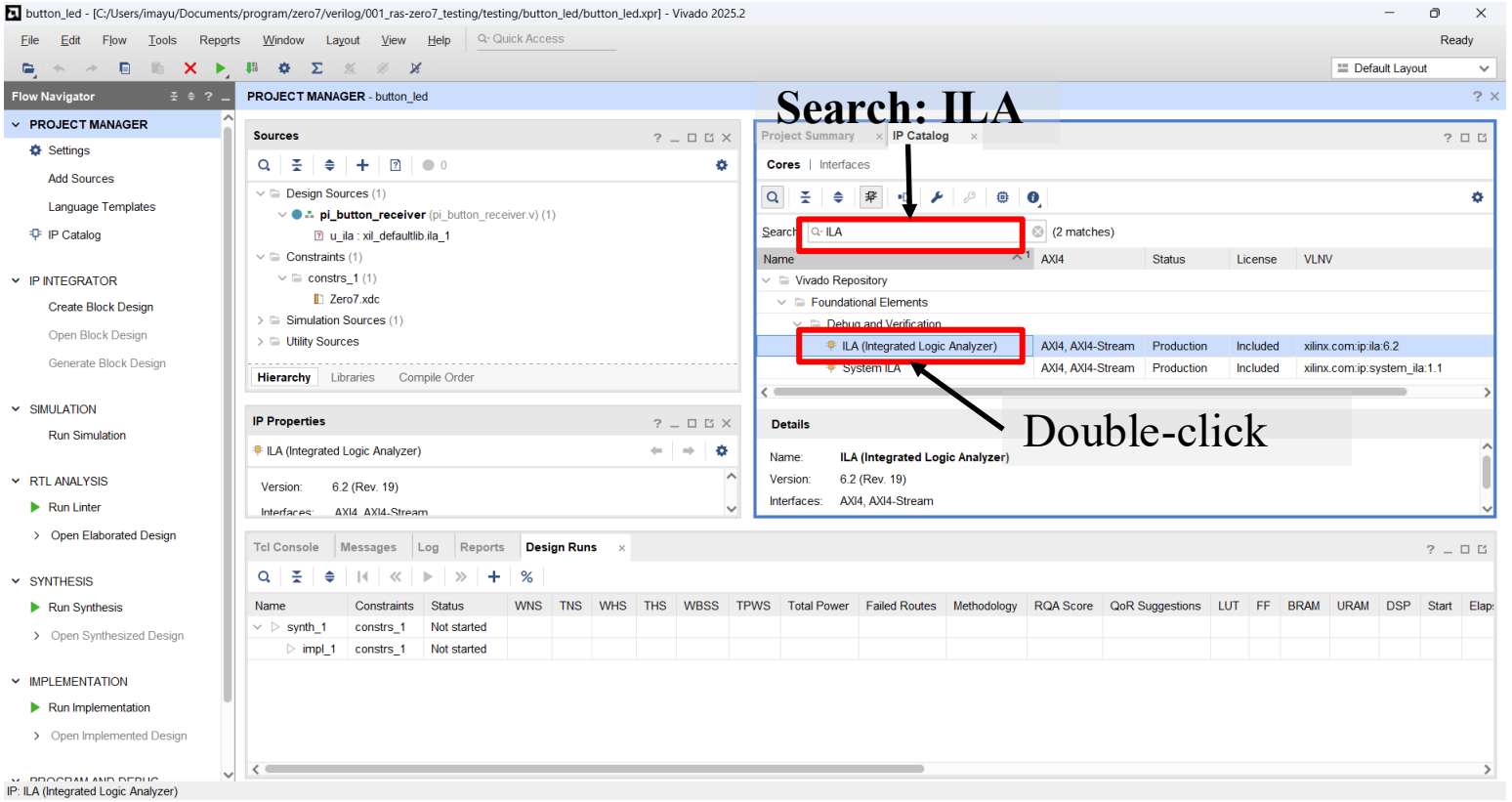
- Add the ILA IP Block
  - Select “IP Catalog”.
  - The “IP Catalog” window opens on the right.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Creating the ILA 3

- Add the ILA IP Block
  - Enter “ILA” in the search box.
  - Double-click “ILA (Integrated Logic Analyzer)”.

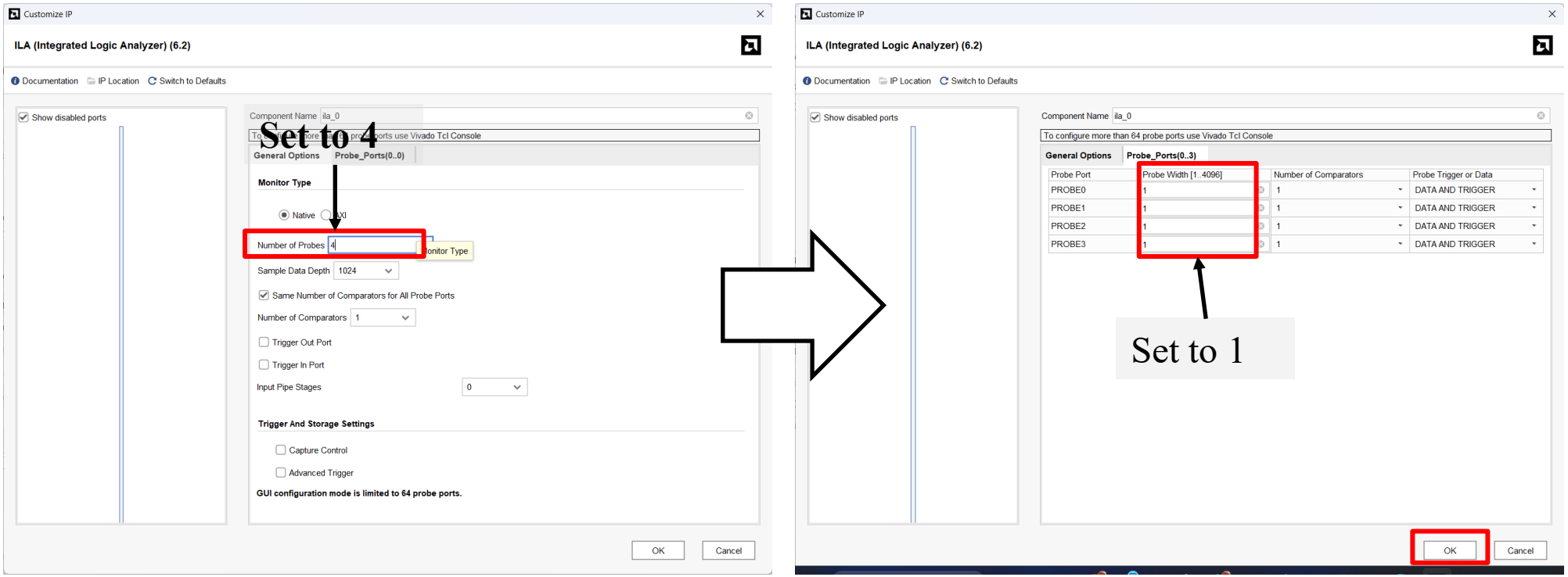


Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Creating the ILA 4

Configure the ILA IP Block

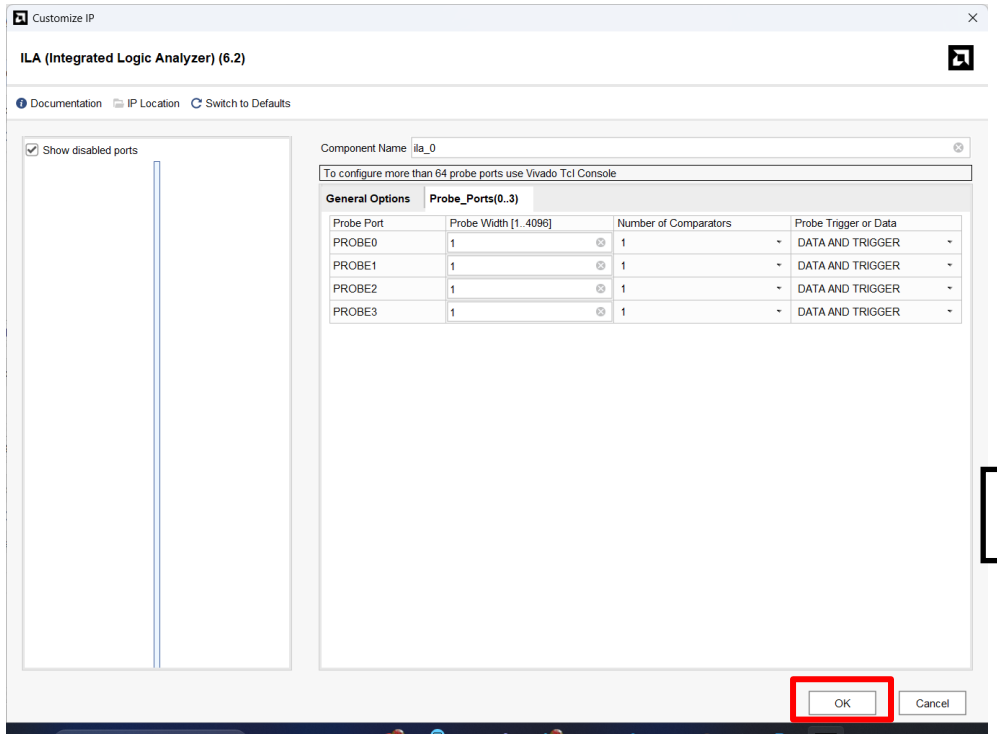
- The ILA configuration window opens.
- Under “General Options”, set “Number of Probes” to 4.
- Set the “Probe Width” of every probe to 1.
- After configuring the settings, click “OK”.

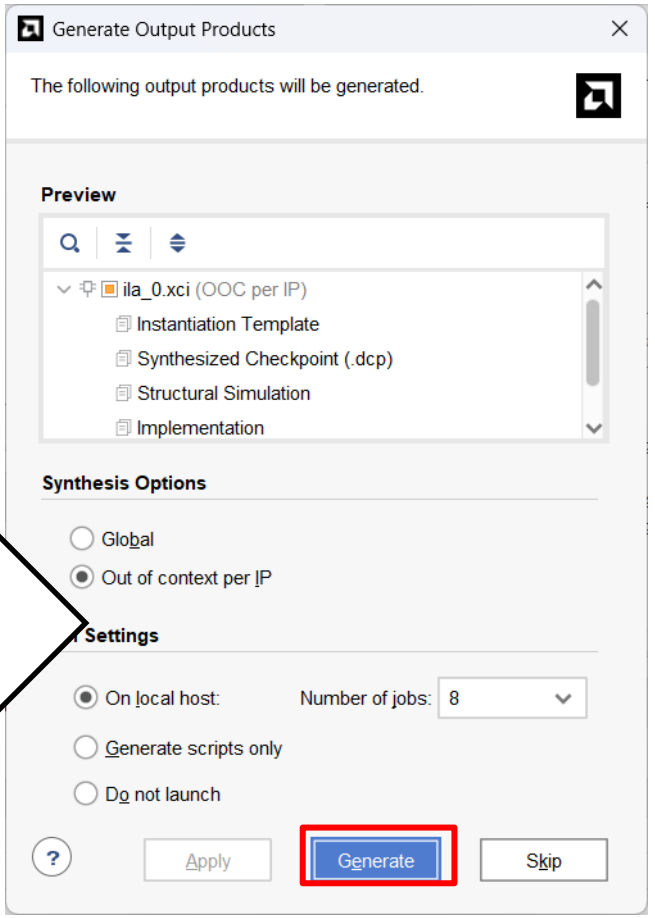


Copyrights 2026 DSP Laboratory. All Rights Reserved.

Generate the ILA IP Block

- After clicking “OK”, the generation window shown below appears.

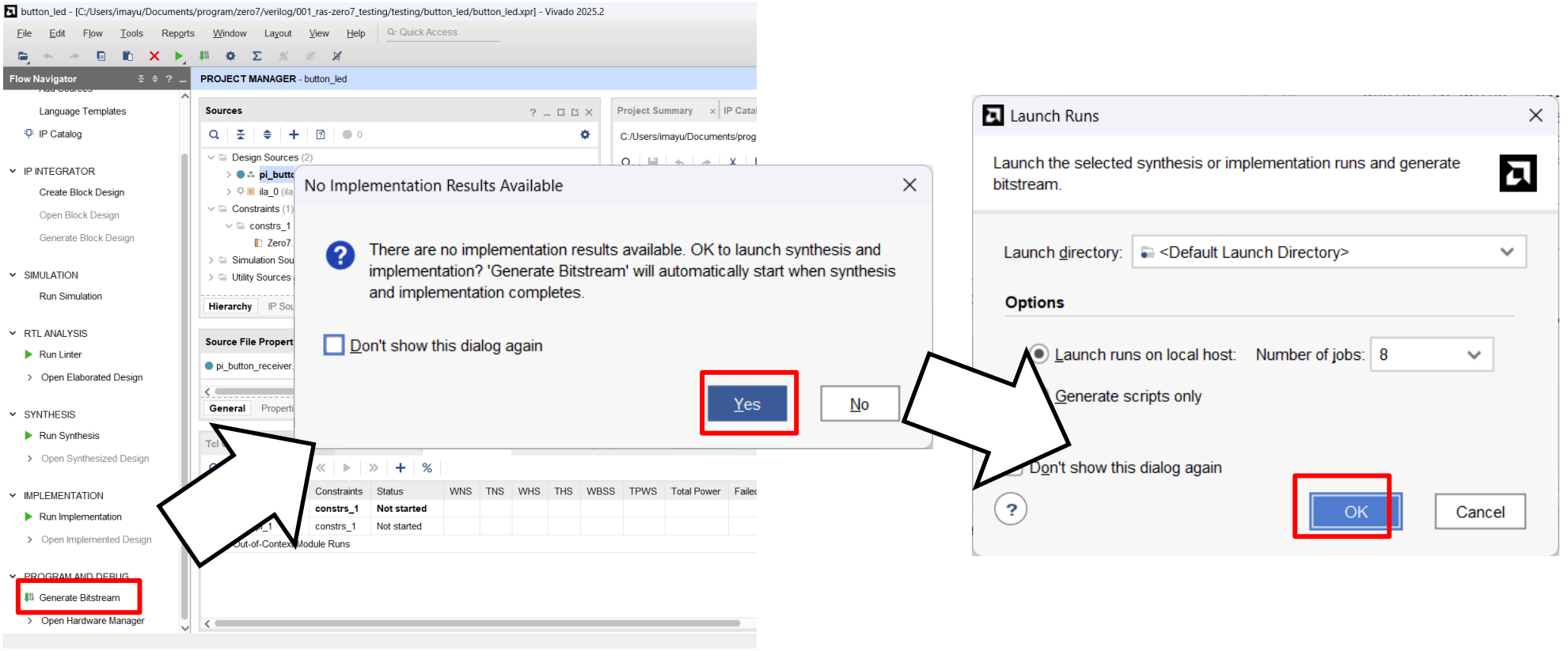




Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Generating the Bitstream 1

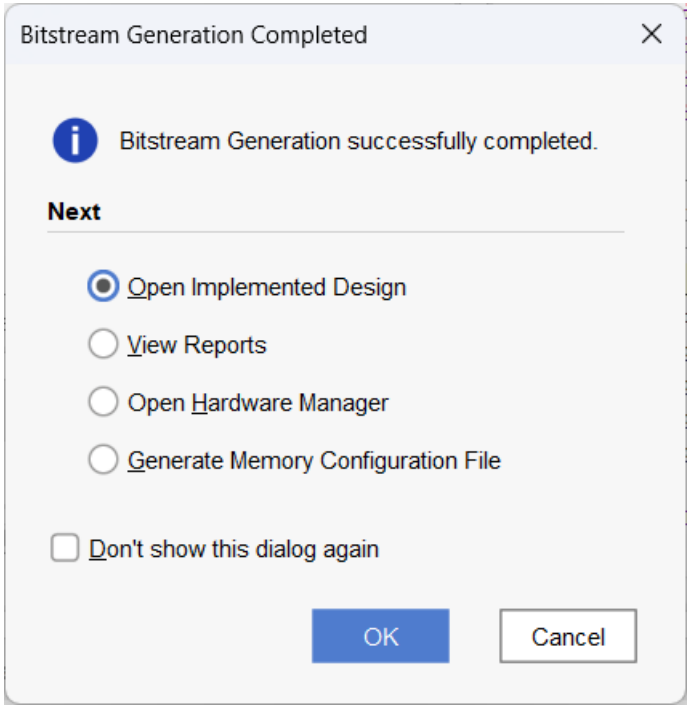
- Select “Generate Bitstream” at the bottom of the left panel.
- Bitstream generation takes some time; wait for it to finish.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Generating the Bitstream 2

- Once the bitstream is generated, the window below appears.
- Click “OK” to display the report for this project.

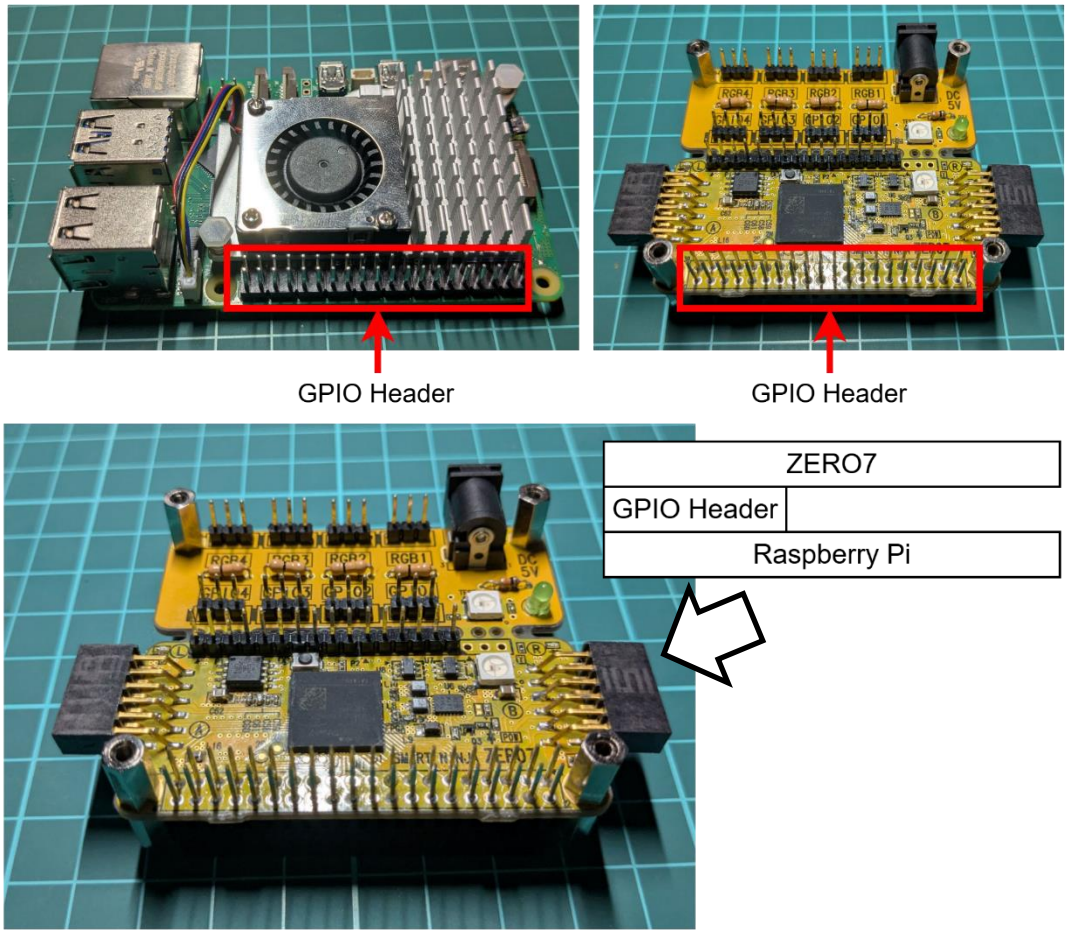


Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Preparing ZERO7 1

- Connect Raspberry Pi and ZERO7 as shown below.
  - Connect their GPIO headers to each other.

- When Raspberry Pi is powered on,
  - the LED on ZERO7 lights up.

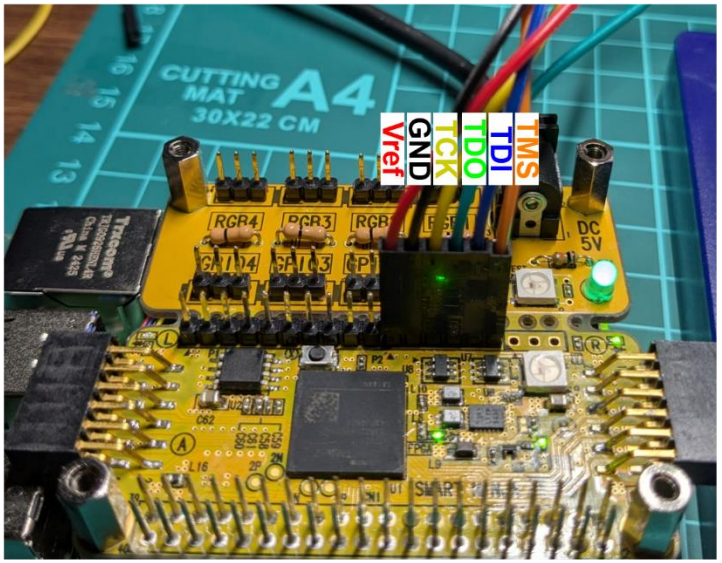


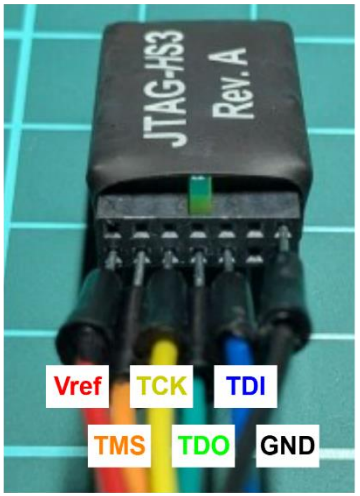
Copyrights 2026 DSP Laboratory. All Rights Reserved.

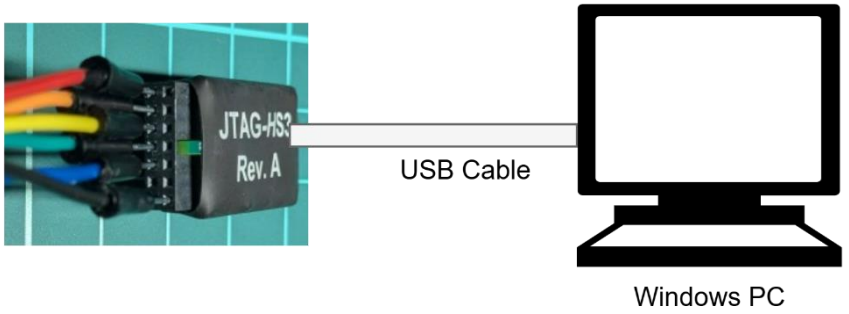
Vivado Project | Connecting ZERO7 to a Windows PC

- Connect the PC and ZERO7 via JTAG
  - Connect JTAG-H53 to the JTAG pins on ZERO7 using jumper wires.
  - Connect JTAG-H53 to the Windows PC using a USB cable.

- Vref (red)
- TMS (orenge)
- TCK (yallow)
- TDO (green)
- TDI (blue)
- GND (black)



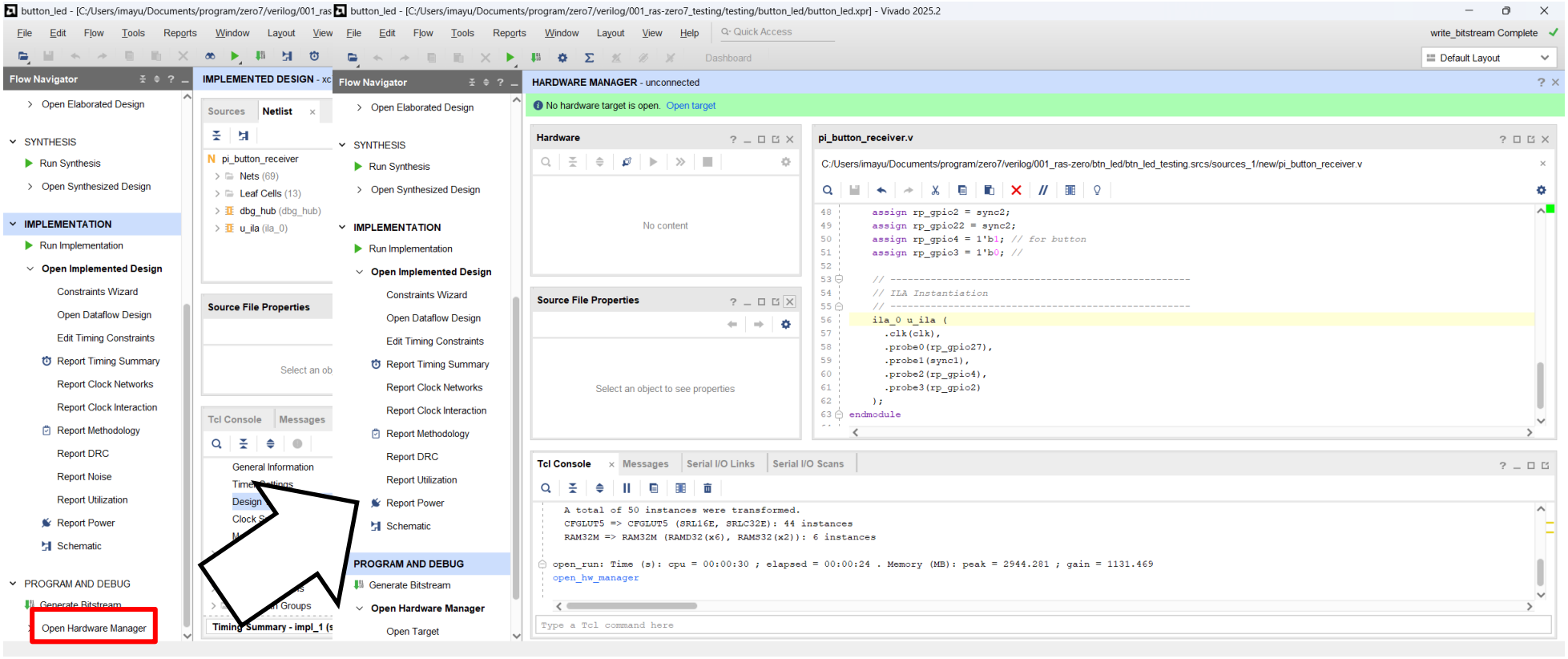




Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Uploading the .bit File 1

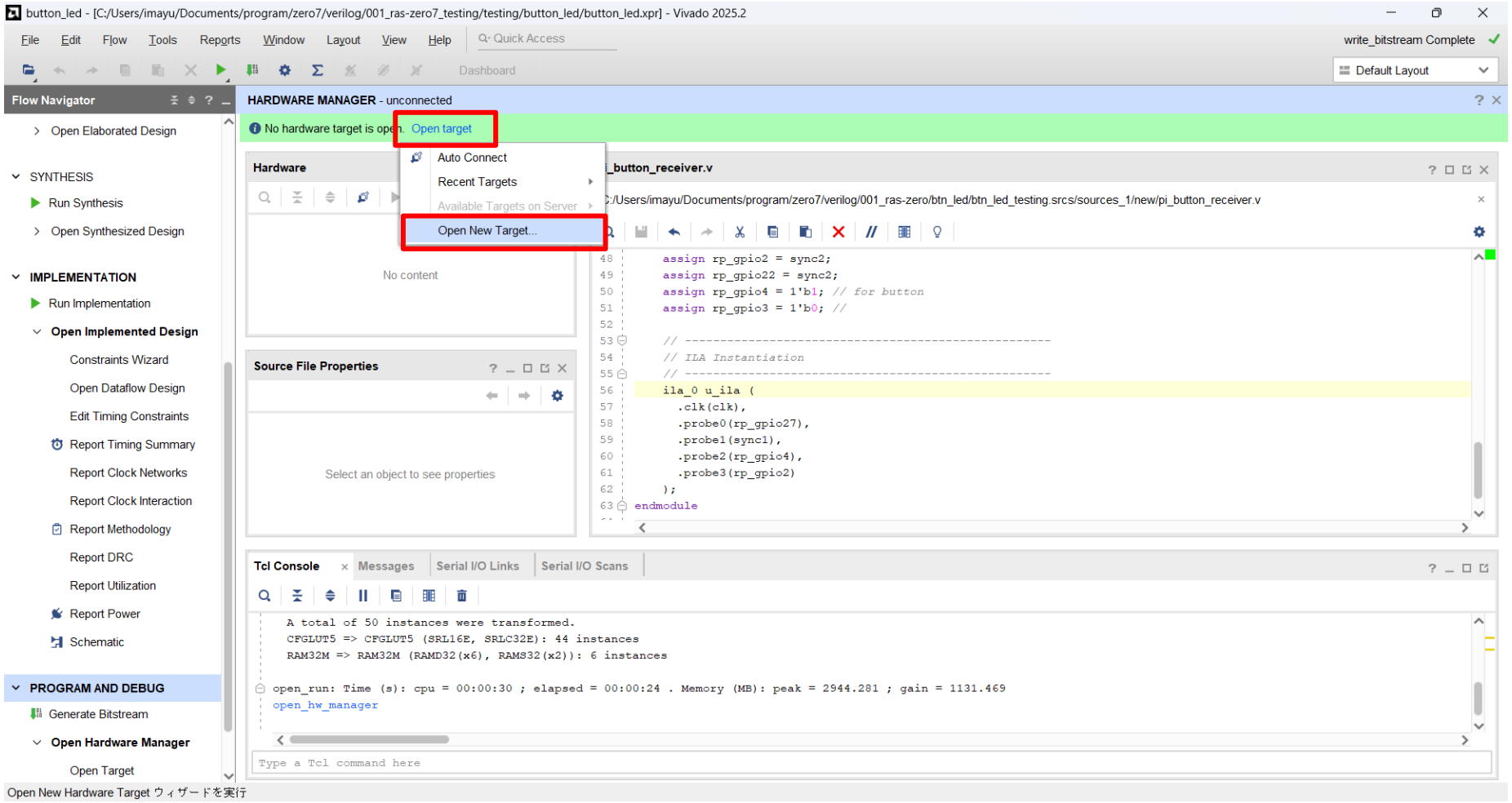
- Connect to ZERO7 via “Open Hardware”
  - Return to Vivado.
  - Select “Open Hardware” below “Generate Bitstream”.
  - “Hardware Manager” opens.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Uploading the .bit File 2

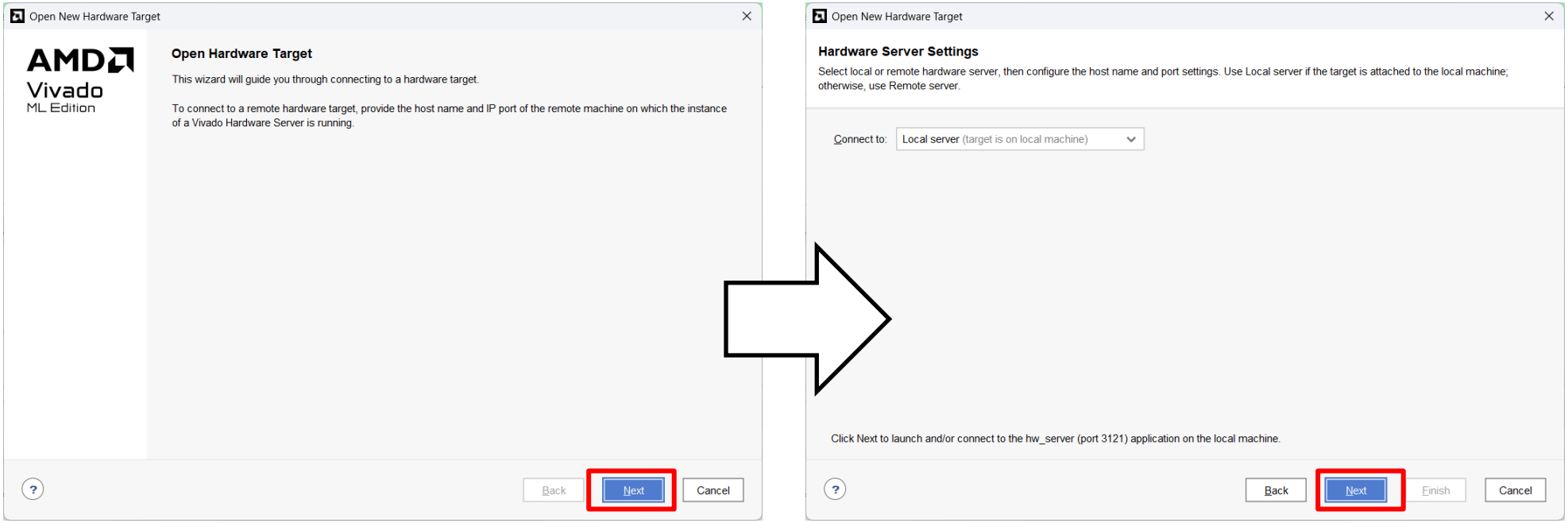
- Connect to ZERO7 via “Open Hardware”
  - Select “Open target”, then “Open New Target”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Uploading the .bit File 3

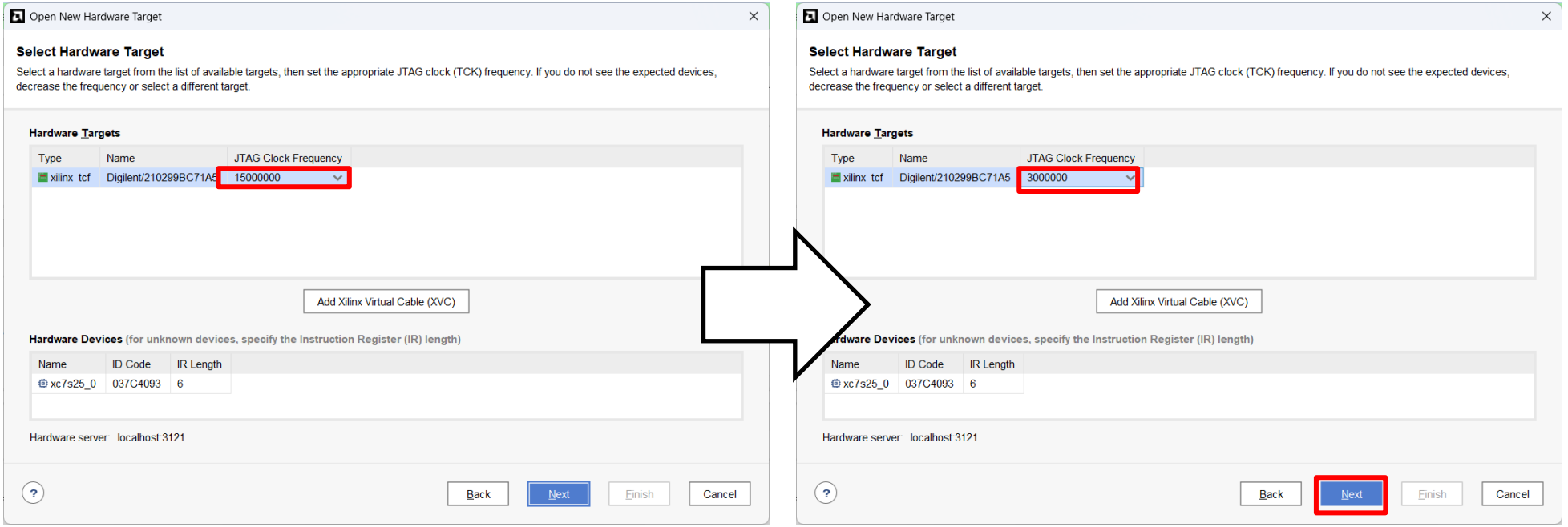
- Connect to ZERO7 via “Open Hardware”
  - After selecting “Open New Target”, the following screen appears.
  - In “Open Hardware Target”, click “Next”.
  - In “Hardware Server Setting”, select “Local server” and click “Next”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Uploading the .bit File 4

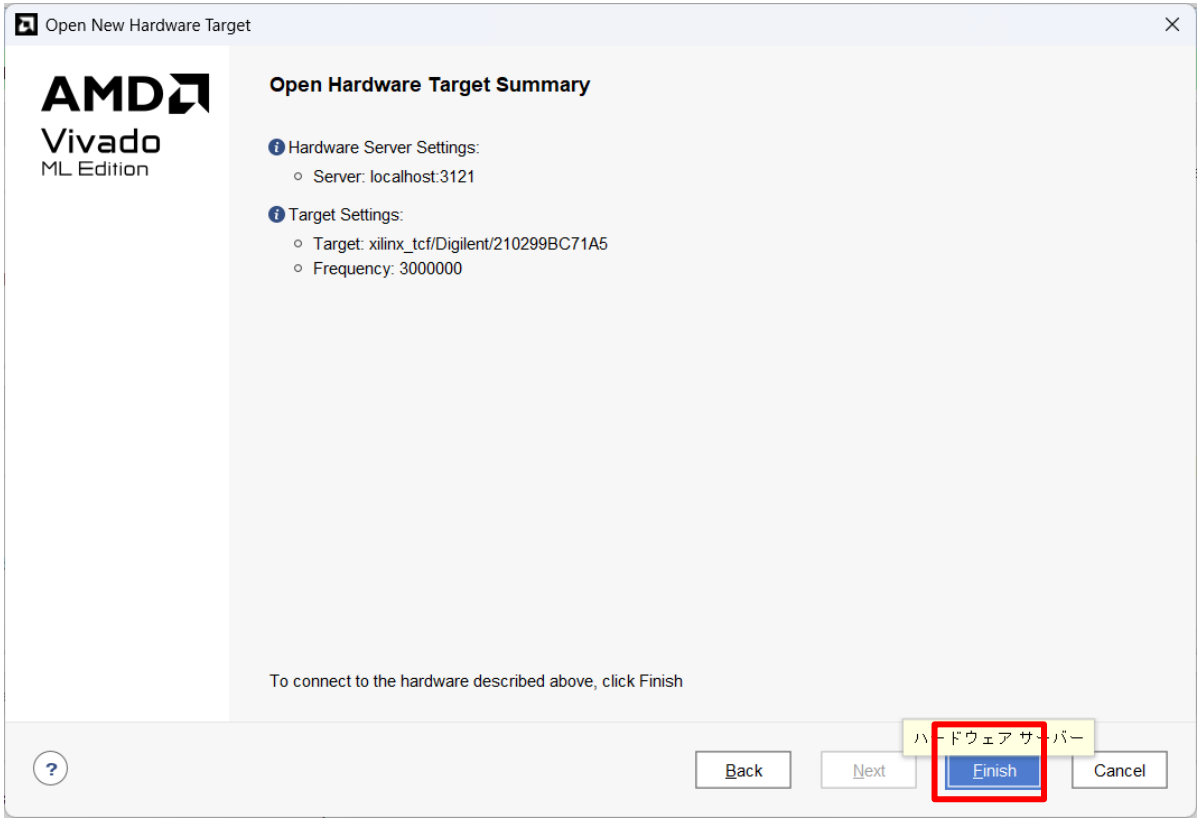
- Connect to ZERO7 via “Open Hardware”
  - In “Setting Hardware Target”, configure the JTAG clock frequency.
  - For stability, set a lower frequency of 3,000,000 Hz.
  - After setting the frequency, click “Next”.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Uploading the .bit File 5

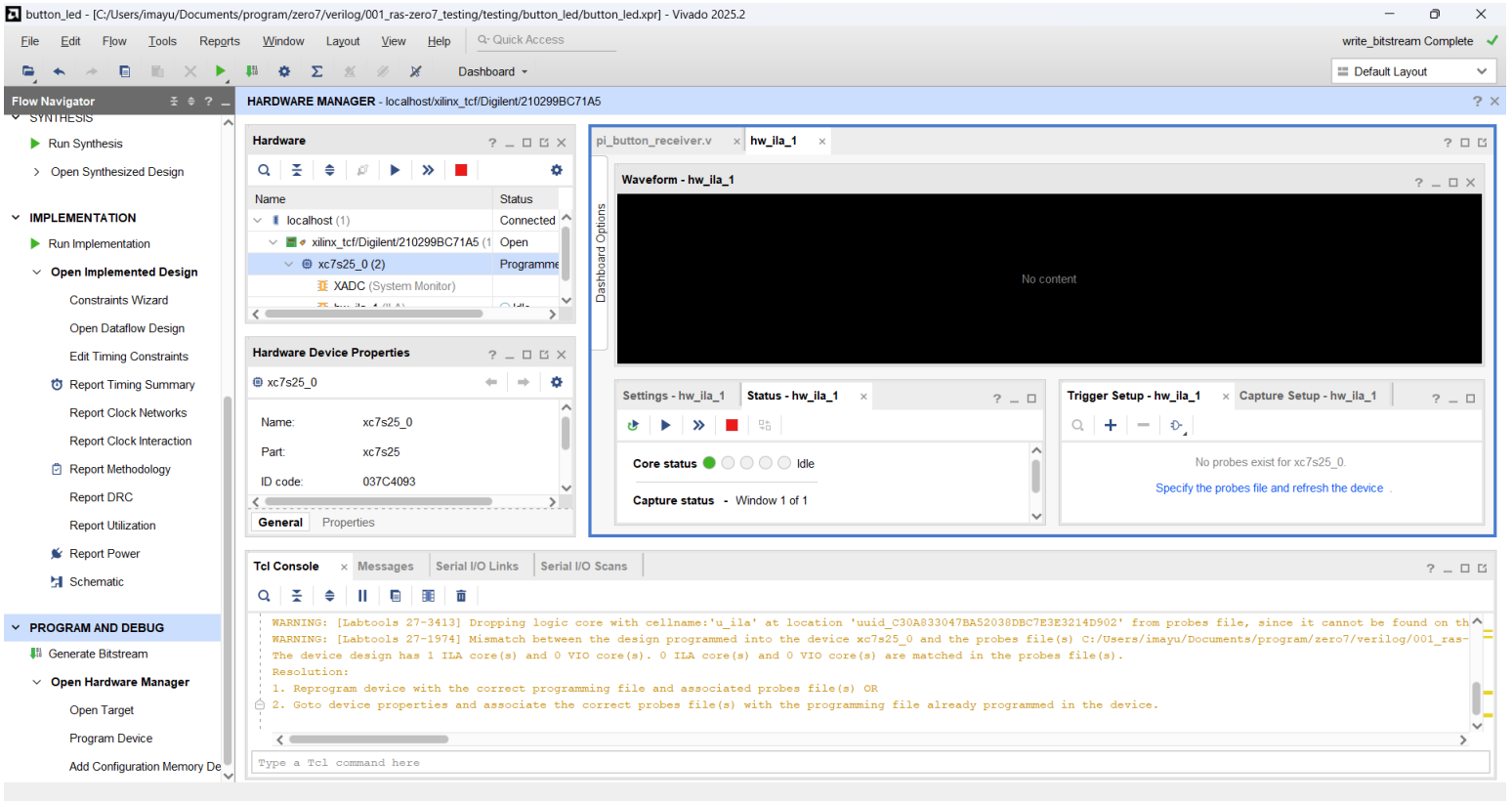
- Connect to ZERO7 via “Open Hardware”
  - If the summary looks like the example below, the connection settings are correct.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Uploading the .bit File 6

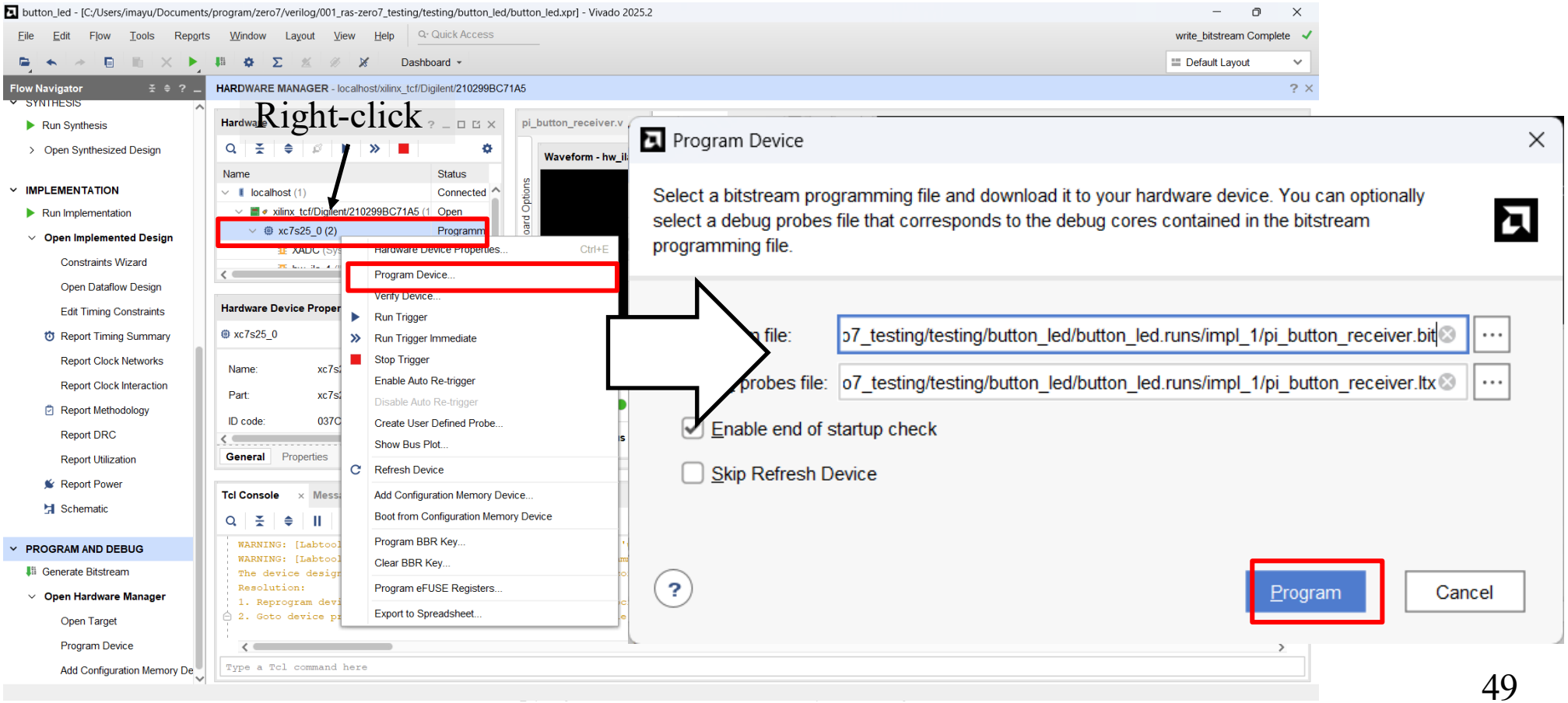
- Verify the Connection
  - If the connection succeeds, the following screen appears.



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Vivado Project | Uploading the .bit File 7

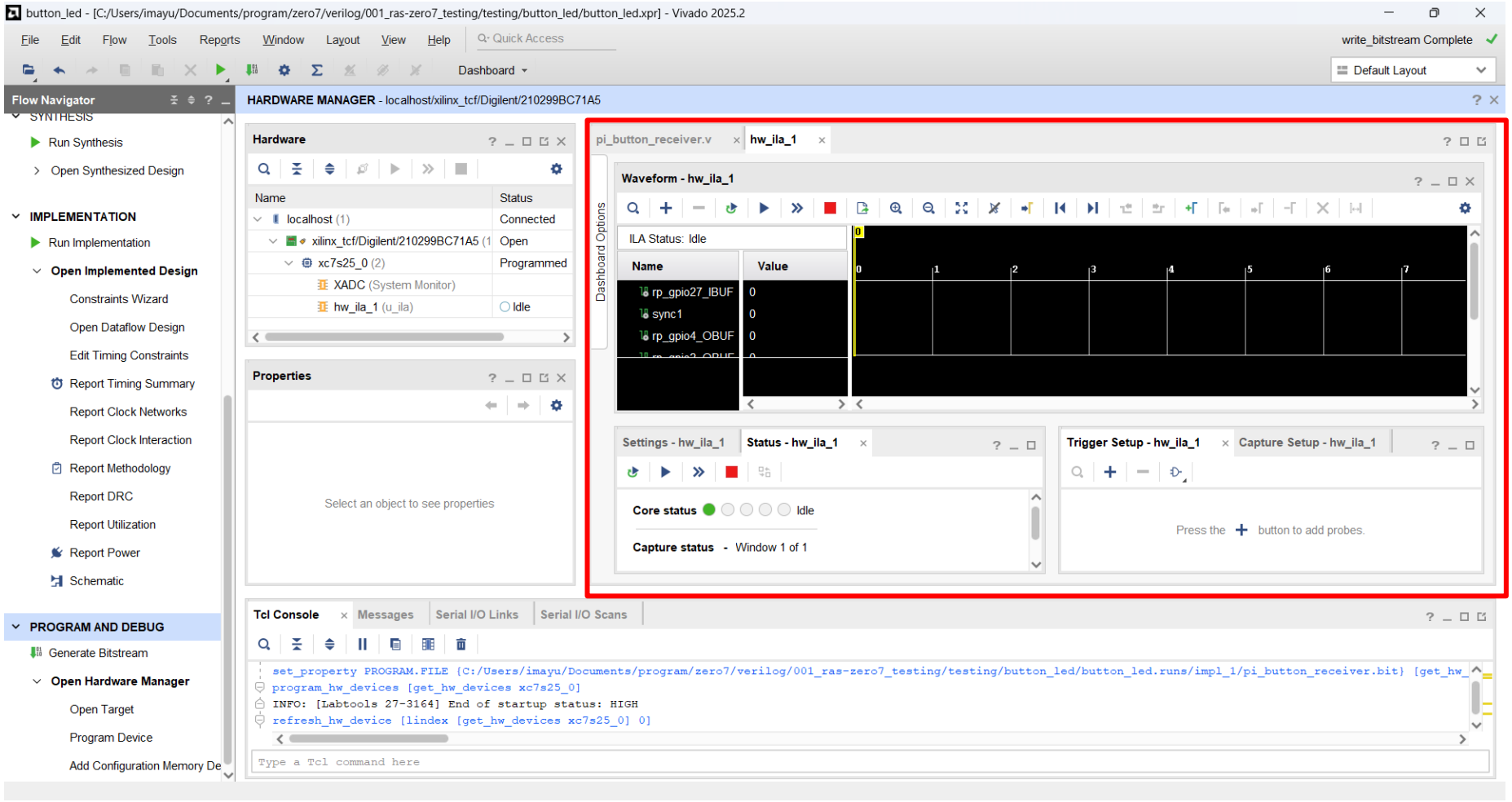
- Upload the .bit File
  - Right-click “xs7s25” → select “Program Device”.
  - Select the .bit file and the ILA probes file (.ltx).
  - Select “Program”.



Vivado Project | Uploading the .bit File 8

Verify the .bit File Upload

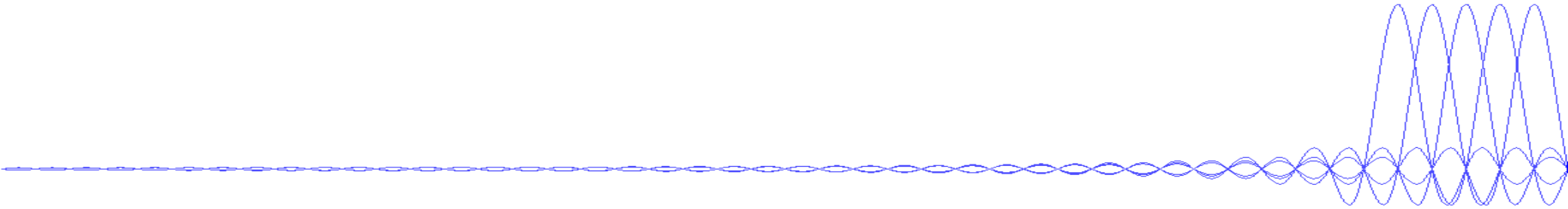
- Confirm that the ILA display has been updated.



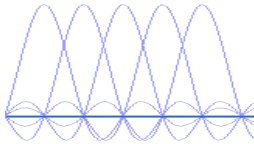
Copyrights 2026 DSP Laboratory. All Rights Reserved.

Raspberry Pi & ZERO7

LED lighting



Copyrights 2026 DSP Laboratory. All Rights Reserved.



Development Workflow

- Build the Circuit
- Run the Program
- Check the ILA

Copyrights 2026 DSP Laboratory. All Rights Reserved.

Specifications of the Example Program

Overview of the Example Program

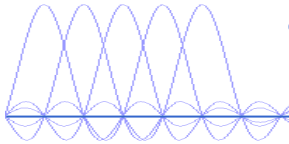
- Raspberry Pi reads a button input and sends the signal to ZERO7.
- ZERO7 receives the signal, applies two-stage synchronization, and turns on the LEDs.

Details 1

- Raspberry Pi
  - Use GPIO 4 for the button input.
  - Console Output
    - Button not pressed → 'False'
    - Button pressed → 'True'
  - Use GPIO 27 to send the signal to ZERO7.
- ZERO7
  - Receive the Raspberry Pi signal through GPIO 27.
  - Synchronize the input through two stages inside the
  - Drive GPIO 2 and GPIO 22 high as outputs.
  - input

input/output	type	variable	Comments
input	wire	clk	
input	wire	rp_gpio27	From Rasp.
output	wire	rp_gpio2	For LED
output	wire	rp_gpio22	For LED
output	wire	rp_gpio3	
output	wire	rp_gpio4	For Button

Copyrights 2026 DSP Laboratory. All Rights Reserved.



Specifications

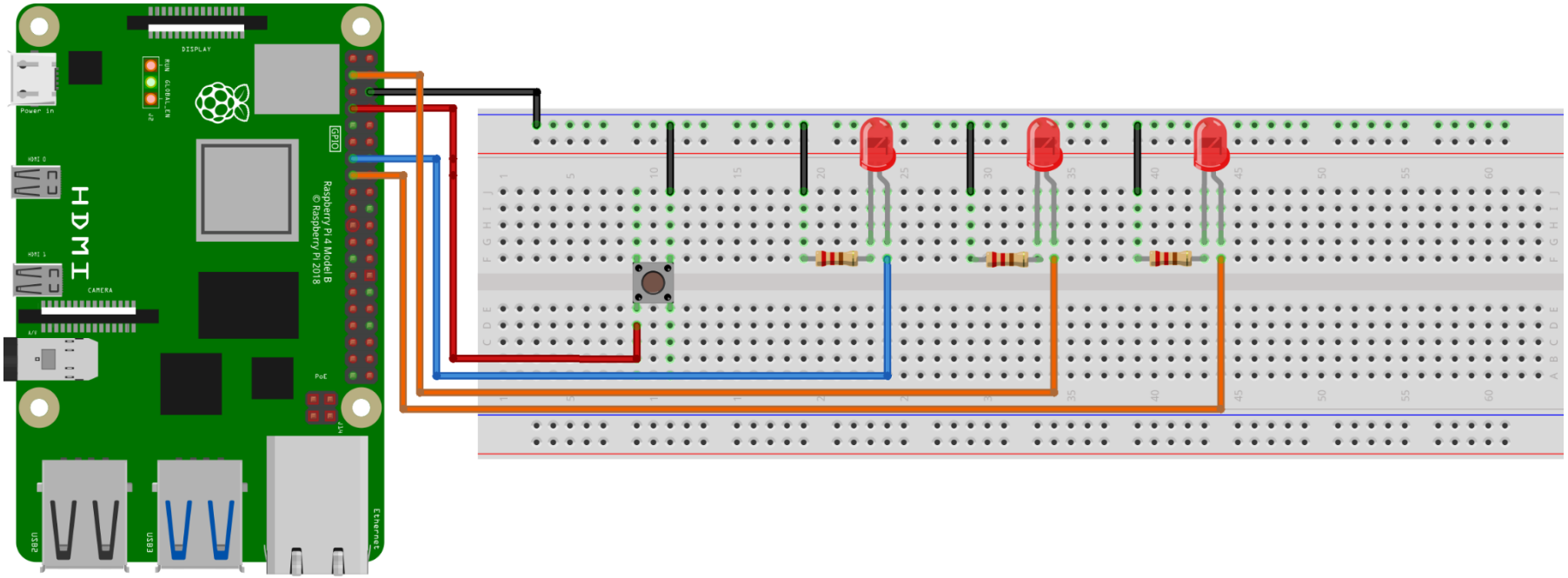
- Details 2
  - DEBUG
    - Use the ILA to monitor input GPIO 27 and output GPIO 22.
    - Connect GPIO 27 to an LED for debugging.

- Example Operation
  - When the button is pressed:
    - Two LEDs driven by ZERO7 light up simultaneously.
    - One additional LED lights up for debugging.

Copyrights 2026 DSP Laboratory. All Rights Reserved.

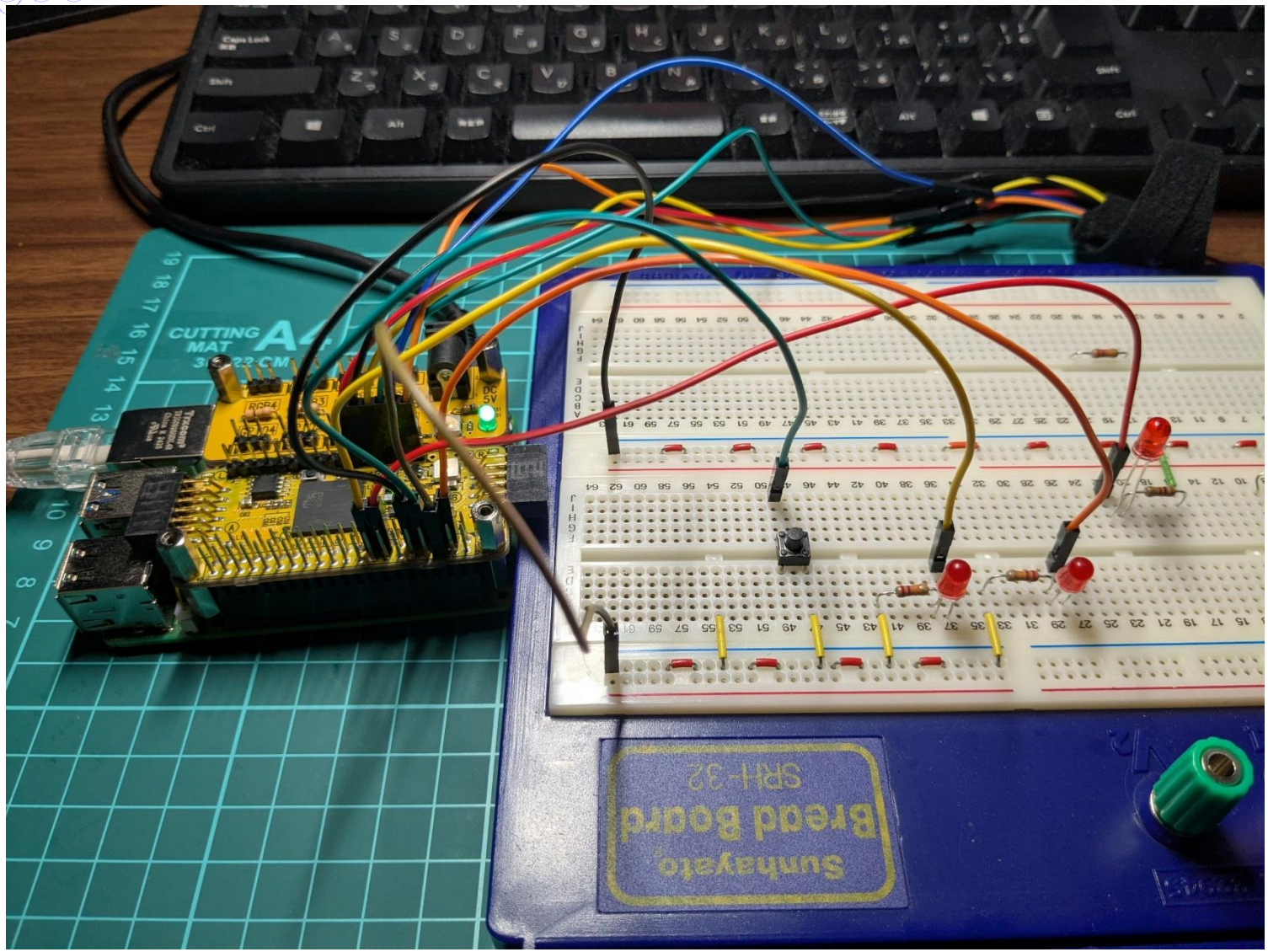
Circuit Diagram

- Connect the breadboard and GPIO pins as shown below.
  - Red wire: Button
  - Orange wire: ZERO7 output
  - Blue wire: Debug signal from Raspberry Pi to ZERO7
  - Black wire: GND



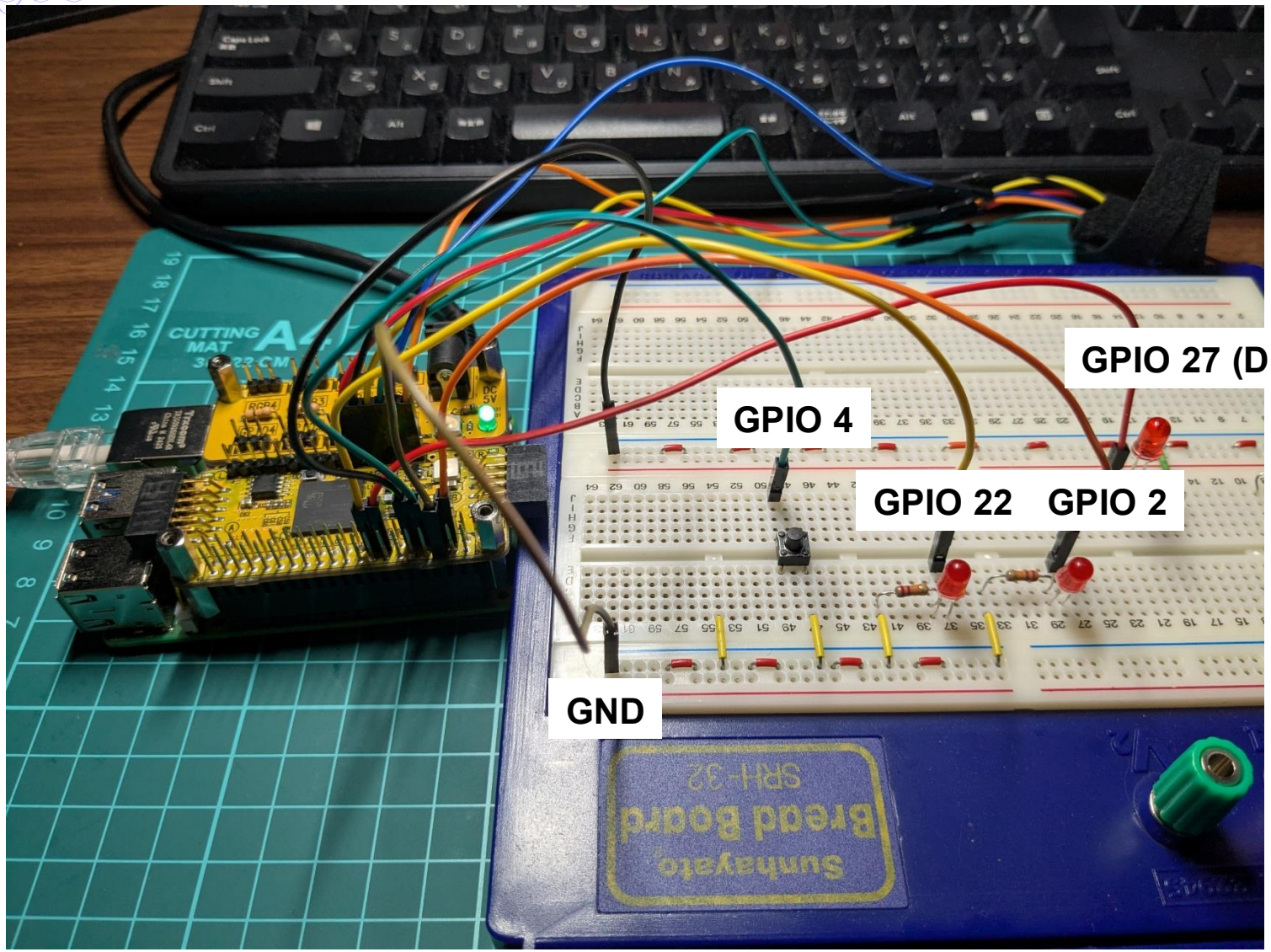
Copyrights 2026 DSP Laboratory. All Rights Reserved.

Circuit Diagram



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Circuit Connections



Copyrights 2026 DSP Laboratory. All Rights Reserved.

Results

- ILA Output
  - rp_gpio27_IBUF: Raspberry Pi input to ZERO7 via GPIO 27
  - rp_gpio2_IBUF: Output from ZERO7 via GPIO 2
  - Due to the two-stage synchronizer, the output is delayed by several clock cycles.

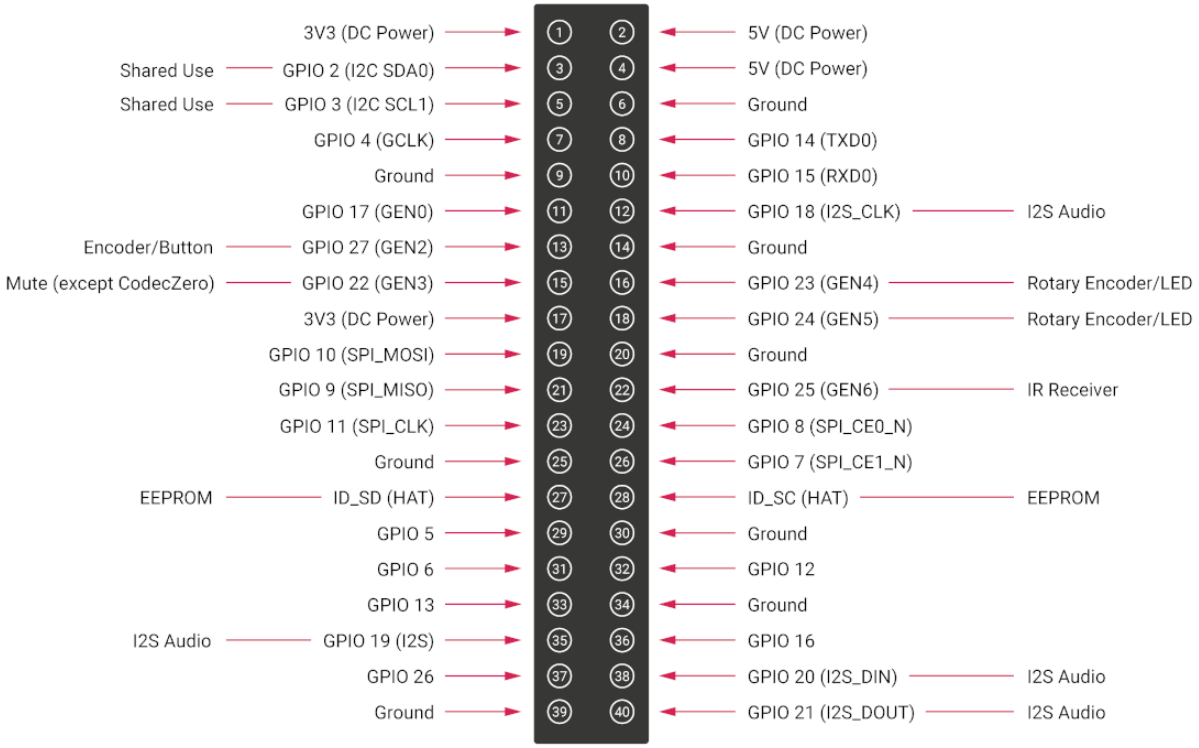


Copyrights 2026 DSP Laboratory. All Rights Reserved.

Appendix: GPIO and Pin Assignments

GPIO Mapping

- The mapping between Raspberry Pi GPIO numbers and physical pin numbers is



Reference: https://www.raspberrypi.com/documentation/accessories/audio.html#gpio-usage

Copyrights 2026 DSP Laboratory. All Rights Reserved.