



Tema 3:

# El entorno de diseño AMD Vivado Design Suite

## Diseño automático de sistemas

**José Manuel Mendías Cuadros**

*Dpto. Arquitectura de Computadores y Automática  
Universidad Complutense de Madrid*

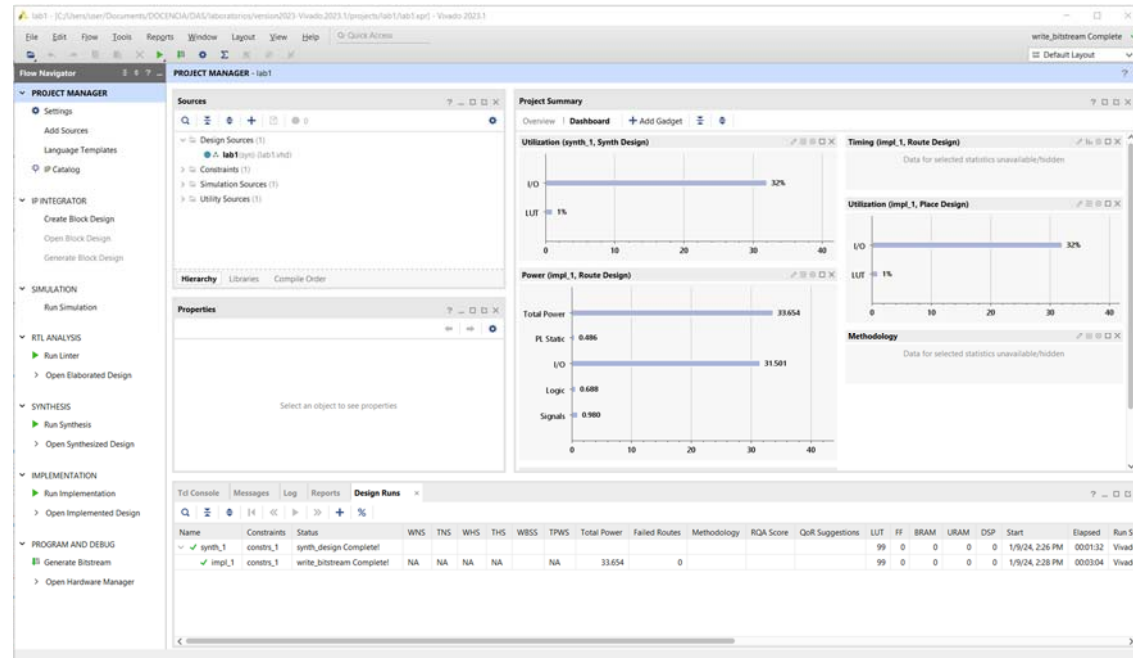


# Contenidos

- ✓ AMD Vivado Design Suite
- ✓ Flujo de diseño y validación.
- ✓ Instalación.
- ✓ Organización de proyectos.
- ✓ Síntesis y prototipado.
- ✓ Análisis de resultados.
- ✓ Validación por simulación.
- ✓ Controlando Vivado.



# AMD Vivado Design Suite

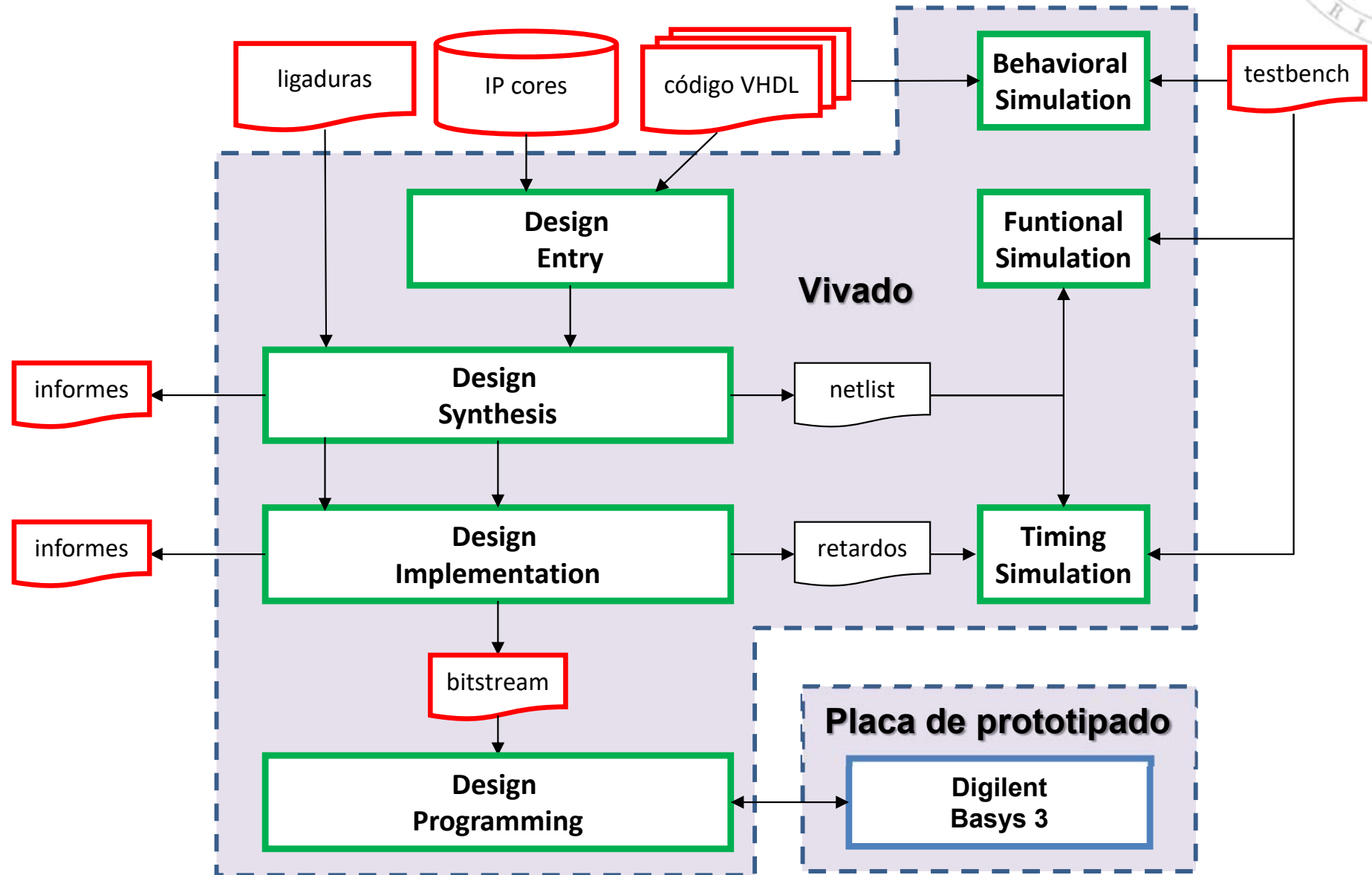


- Conjunto de herramientas para la **síntesis lógica y física** de sistemas digitales sobre **FPGAs** integradas en un interfaz gráfico común:
  - o Gestión integral del proyecto.
  - o Edición del código fuente.
  - o Síntesis.
  - o Validación por simulación.
  - o Volcado de ficheros de configuración.



# AMD Vivado Design Suite

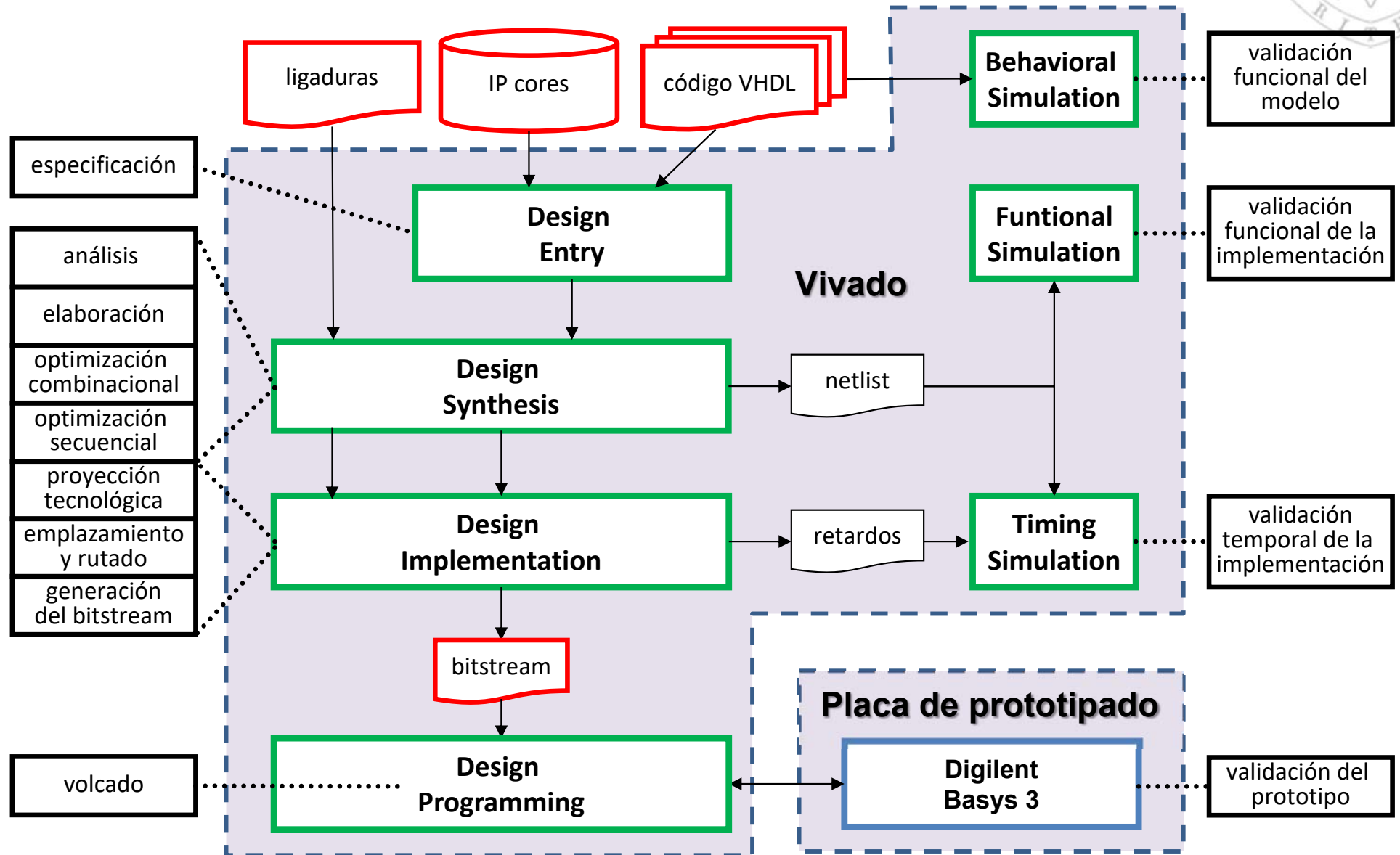
## flujo de diseño y validación (i)





# AMD Vivado Design Suite

## flujo de diseño y validación (ii)



# AMD Vivado Design Suite

## flujo de diseño y validación (iii)



- Las **ligaduras (constraints)** permiten:
  - Controlar el ciclo de diseño e implementación.
  - Fijar los niveles mínimos de calidad aceptables de una implementación.
  - Describir las características del resto del sistema en donde se integrará la FPGA.
  
- Las ligaduras pueden clasificarse según definan:
  - **Técnicas de diseño a aplicar** (locales a módulo):
    - Diseño jerárquico, codificación de FSM, estilo de RAM (block o distribuida) ...
  - **Estrategias de optimización** (aplicables a todo el diseño):
    - Objetivo de la optimización (área o velocidad), esfuerzo (normal o alto) ...
  - **Ligaduras físicas**:
    - Tiempo de ciclo, retardos, localización (celdas y pines) ...
  
- Las ligaduras pueden establecerse a través de:
  - **GUI**: método simple.
  - **Atributos de HDL**: método asociable a HDL.
  - **Fichero de ligaduras** (fichero XDC): método más compacto.

# AMD Vivado Design Suite

## flujo de diseño y validación (iv)



- Los **informes (reports)** son ficheros generados por las herramientas que contienen información sobre los procesos efectuados:
  - Avisos y errores encontrados durante el proceso de diseño e implementación.
  - Herramientas usadas y opciones utilizadas.
  - Decisiones de diseño tomadas (inferencias).
  - Estadísticas de caracterización de la implementación.
- La visualización de informes puede ser:
  - **Directa** (fichero texto).
  - A través de **GUI**.

# AMD Vivado ML Edition

## instalación (i)



- Descargar y ejecutar la versión del instalador Vivado ML Edition - 2023.1 desde la página: <https://www.xilinx.com/support/download.html>
  - Para la descarga solicitará iniciar sesión, por ello deberá crearse previamente una cuenta en AMD

 AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1: Windows Self Extracting Web Installer (EXE - 199.47 MB)

MD5 SUM Value : 4c6a1e5d5cf7c44c3f201c9056b6cf45

Download Verification 

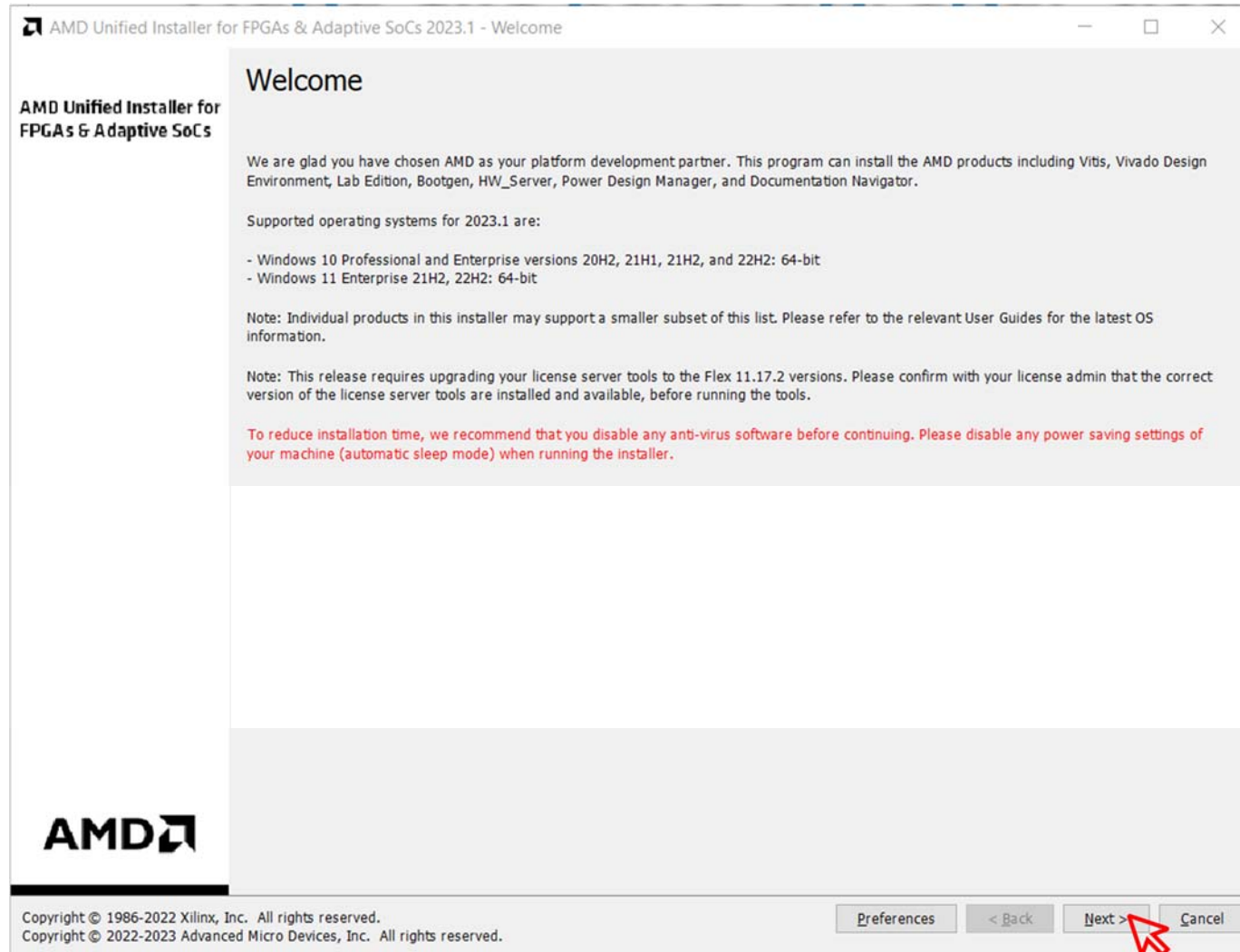
Digests

Signature

Public Key

# AMD Vivado ML Edition

## instalación (ii)



# AMD Vivado ML Edition

## instalación (ii)



AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1 - Select Install Type

**Select Install Type**

Please select install type and provide your AMD.com E-mail Address and password for authentication.

**User Authentication**

Please provide your AMD user account credentials to download the required files.  
If you don't have an account, [please create one](#). If you forgot your password, you can [reset it here](#).

E-mail Address

Password

☒ **Download and Install Now**

Select your desired device and tool installation options and the installer will download and install just what is required.

☐ **Download Image (Install Separately)**

The installer will download an image containing all devices and tool options for later installation. Use this option if you wish to install a full image on a network drive or allow different users maximum flexibility when installing.

Copyright © 1986-2022 Xilinx, Inc. All rights reserved.  
Copyright © 2022-2023 Advanced Micro Devices, Inc. All rights reserved.

< Back Next > Cancel

# AMD Vivado ML Edition

## instalación (iii)



AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1 - Select Product to Install

**Select Product to Install**

Select a product to continue installation. You will be able to customize the content in the next page.

☐ Vitis

Installs Vitis Core Development Kit for embedded software and application acceleration development on AMD platforms. Vitis installation includes Vivado Design Suite. Users can also install Vitis Model Composer to design for AI Engines and Programmable Logic in MATLAB and Simulink.

☒ **Vivado**

Includes the full complement of Vivado Design Suite tools for design, including C-based design with Vitis High-Level Synthesis, implementation, verification and device programming. Complete device support, cable driver, and Document Navigator included. Users can also install Vitis Model Composer to design for AI Engines and Programmable Logic in MATLAB and Simulink.

☐ BootGen

Installs Bootgen for creating bootable images targeting AMD SoCs and FPGAs.

☐ Lab Edition

Installs only the Vivado Lab Edition. This standalone product includes Vivado Design Programmer, Vivado Logic Analyzer and UpdateMEM tools.

☐ Hardware Server

Installs hardware server and JTAG cable drivers for remote debugging.

☐ Power Design Manager (PDM)

Installs only the Power Design Manager (PDM). Power Design Manager is a standalone design tool used to estimate power requirements of Versal and Kria products. It supports the Xilinx Power Estimator (XPE) file exchange format for importing data from Vivado and XPE.

☐ Documentation Navigator (Standalone)

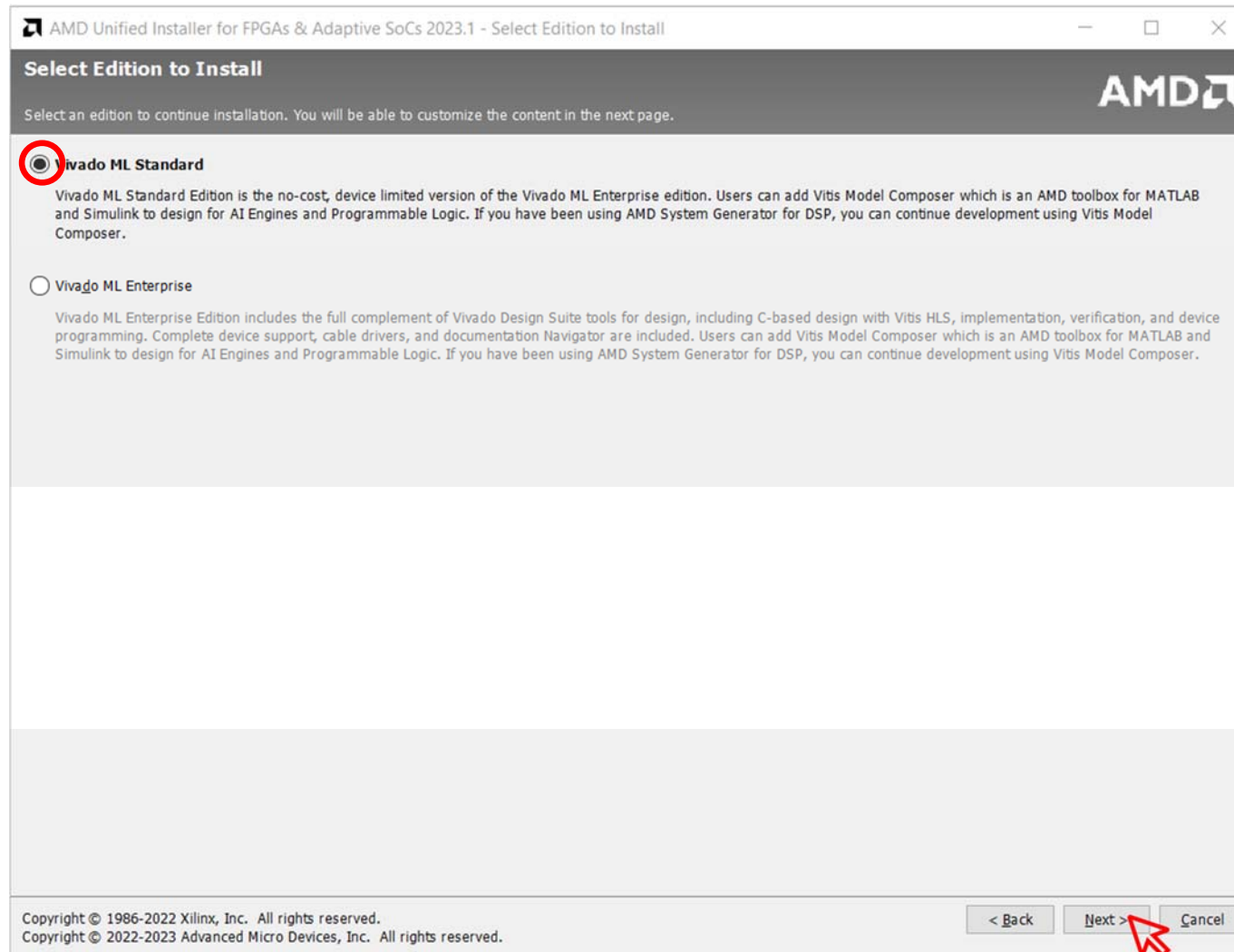
Documentation Navigator (DocNav) provides access to AMD FPGAs & Adaptive SoCs technical documentation both on the Web and on the Desktop. This is a standalone installation without Vivado Design Suite.

Copyright © 1986-2022 Xilinx, Inc. All rights reserved.  
Copyright © 2022-2023 Advanced Micro Devices, Inc. All rights reserved.

< Back Next > Cancel

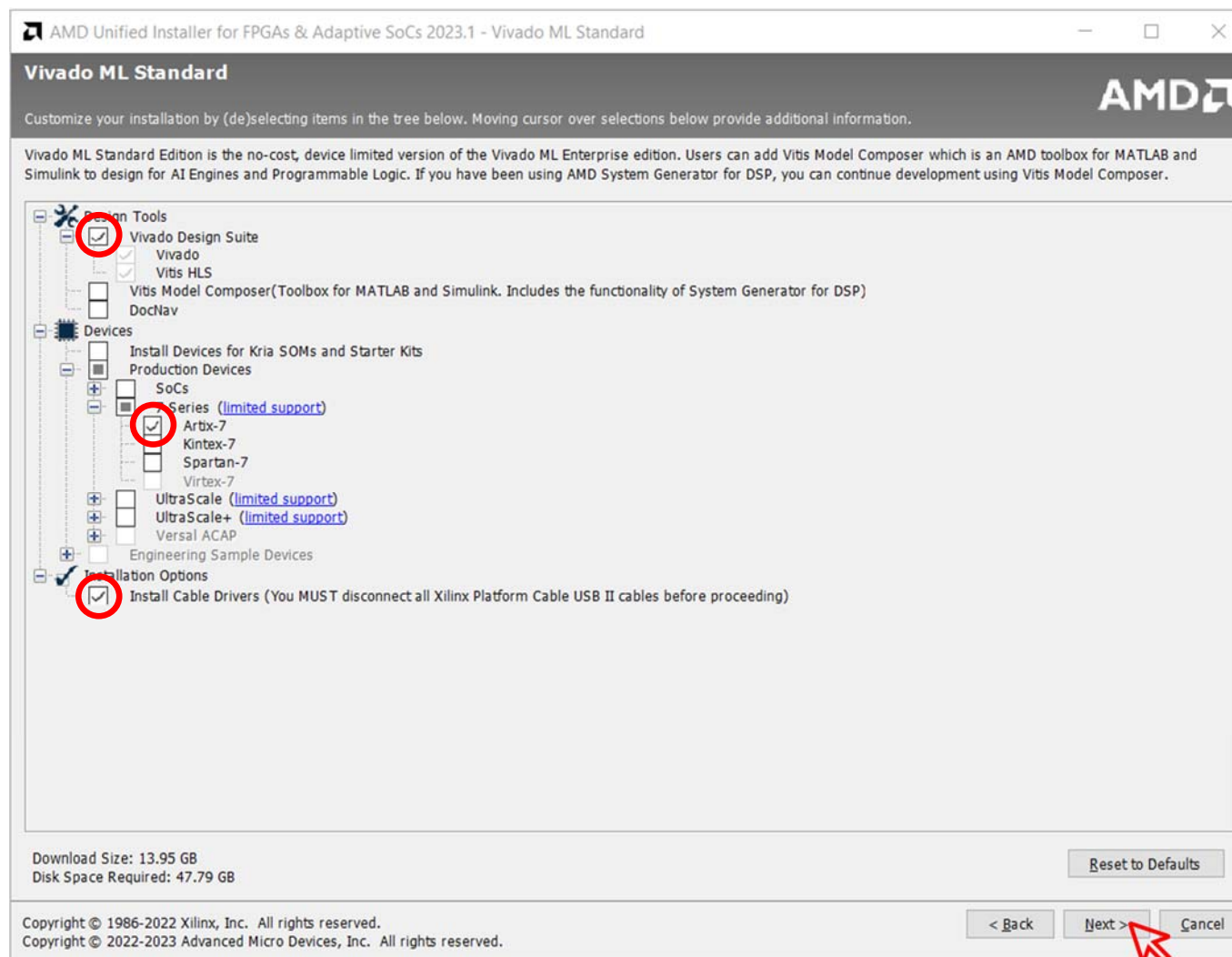
# AMD Vivado ML Edition

## instalación (iv)



# AMD Vivado ML Edition

## instalación (v)



# AMD Vivado ML Edition

## instalación (vi)



AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1 - Accept License Agreements

### Accept License Agreements

Please read the following terms and conditions and indicate that you agree by checking the I Agree checkboxes.

**End User License Agreement for Vivado**

By checking "I Agree" below, or OTHERWISE ACCESSING, DOWNLOADING, INSTALLING or USING THE SOFTWARE, I AGREE on behalf of licensee to be bound by the agreement, which can be viewed by [clicking here](#).

☒ Agree

**Third Party Software End User License Agreement for Vivado**

By checking "I AGREE" below, or OTHERWISE ACCESSING, DOWNLOADING, INSTALLING or USING THE SOFTWARE, YOU AGREE on behalf of licensee to be bound by the agreement, which can be viewed by [clicking here](#).

☒ Agree

Copyright © 1986-2022 Xilinx, Inc. All rights reserved.  
Copyright © 2022-2023 Advanced Micro Devices, Inc. All rights reserved.

< Back   Next >   Cancel

# AMD Vivado ML Edition

## instalación (vii)



AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1 - Select Destination Directory

### Select Destination Directory

Choose installation options such as location and shortcuts.

**Installation Options**

Select the installation directory

C:\Xilinx

**Installation location(s)**

C:\Xilinx\Vivado/2023.1

C:\Xilinx\Vitis\_HLS/2023.1

**Download location**

C:\Xilinx\Downloads\Vivado\_2023.1

**Disk Space Required**

|                       |           |
|-----------------------|-----------|
| Download Size:        | 13.95 GB  |
| Disk Space Required:  | 47.79 GB  |
| Final Disk Usage:     | 29.32 GB  |
| Disk Space Available: | 746.95 GB |

**Select shortcut and file association options**

☒ Create program group entries

Xilinx Design Tools

☒ Create desktop shortcuts

☒ Create file associations

Apply shortcut & file association selections to

☐ Current user

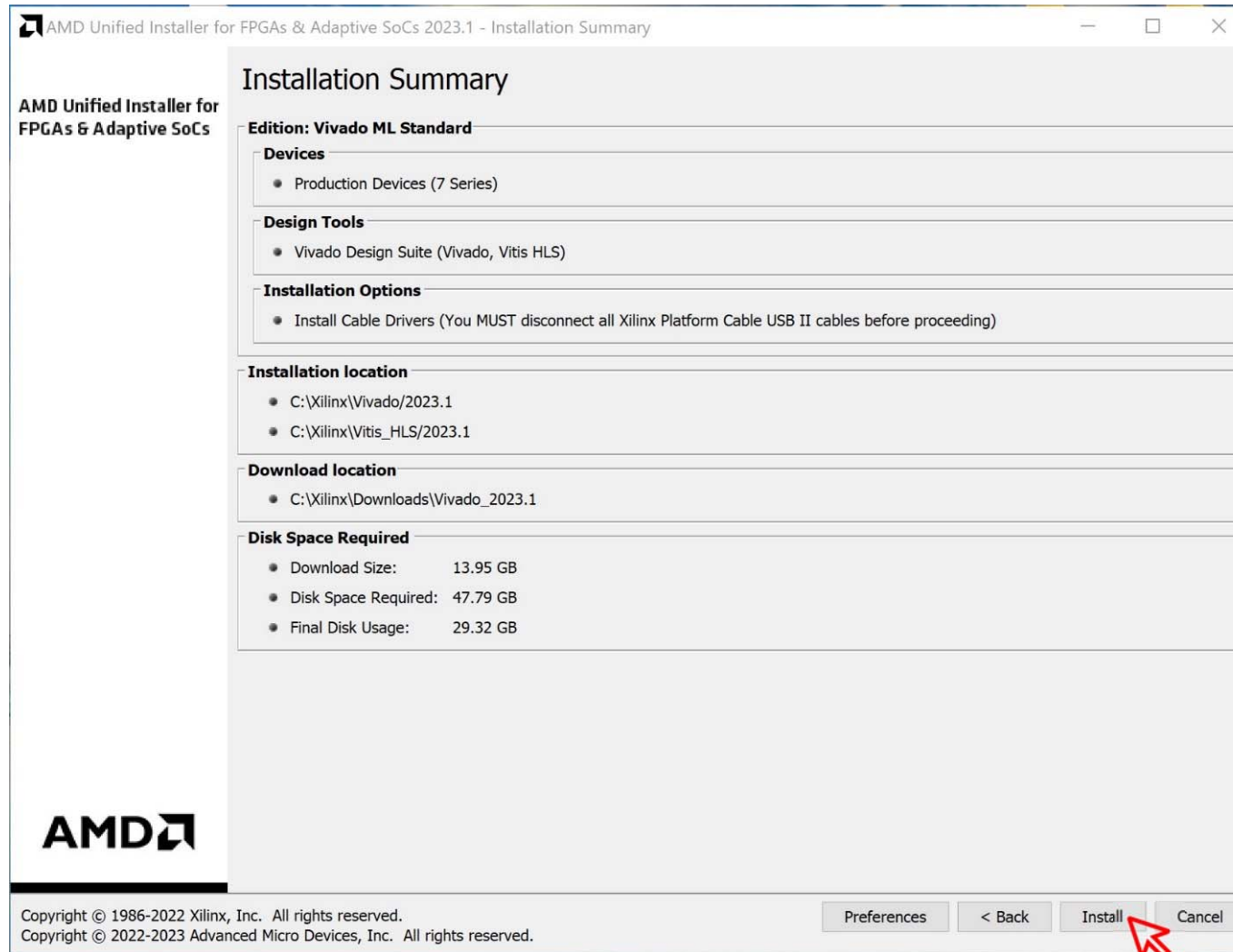
☒ All users

Copyright © 1986-2022 Xilinx, Inc. All rights reserved.  
Copyright © 2022-2023 Advanced Micro Devices, Inc. All rights reserved.

< Back Next > Cancel

# AMD Vivado ML Edition

## instalación (viii)

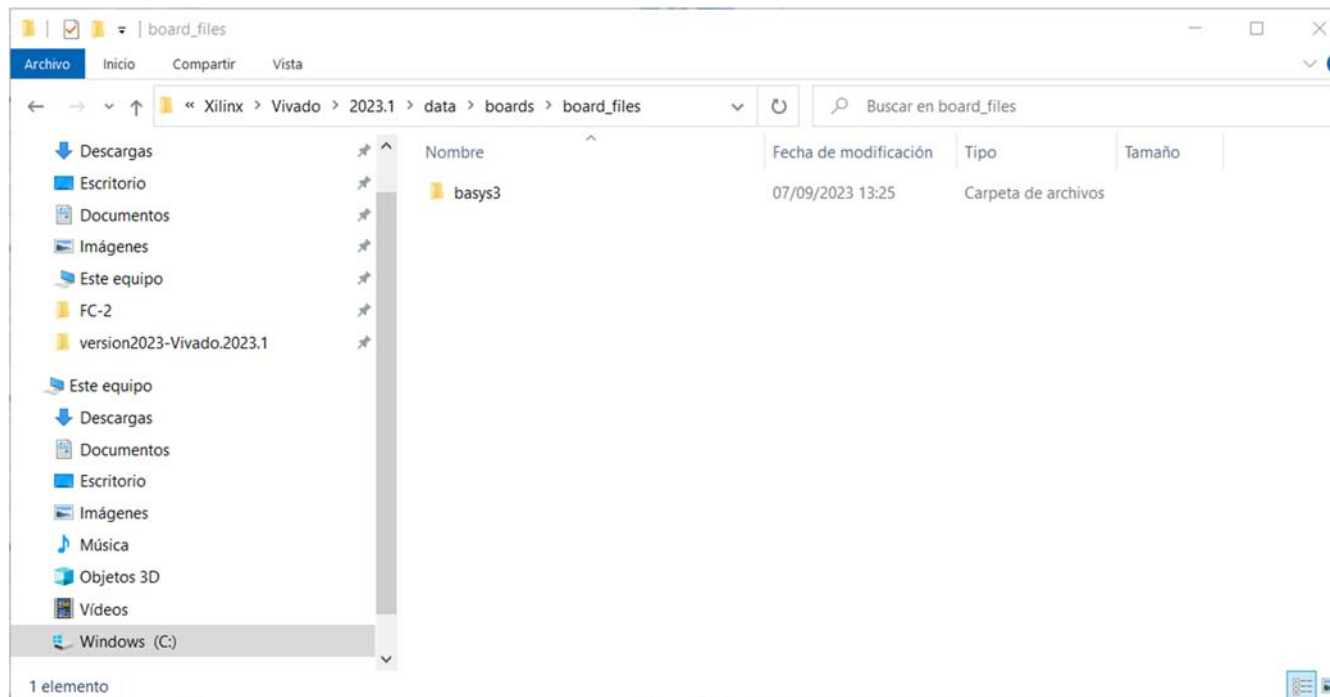


# AMD Vivado ML Edition

## instalación (ix)



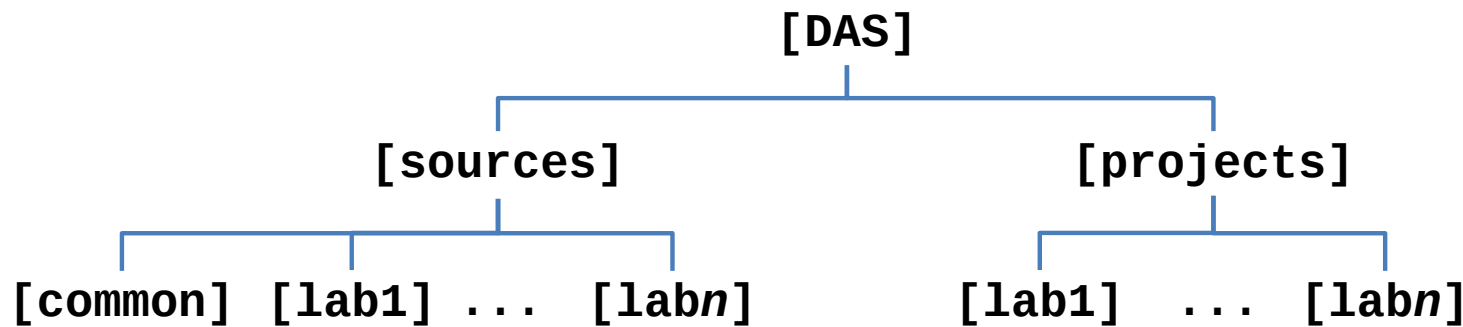
- Descargar de la web de la asignatura el fichero de definición de propiedades de la placa Basys3: **basys3-board-master.zip**
- En el directorio **C:\Xilinx\Vivado\2023.1\data\boards** crear la carpeta **board\_files**
- Copiar en ella la carpeta **basys3** incluida en el fichero descargado.





# Organización de proyectos

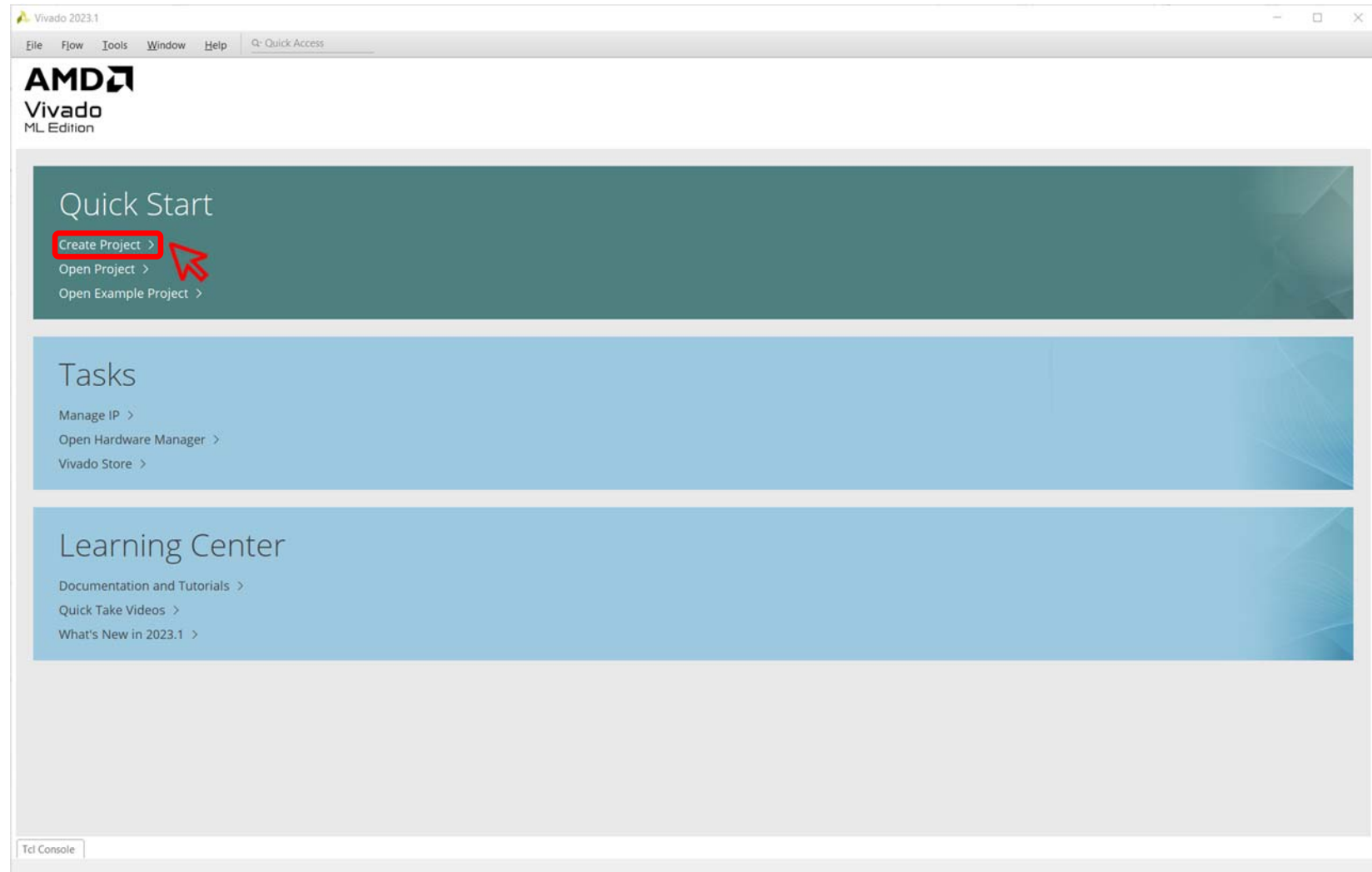
- La **estructura de directorios** de nuestro entorno de trabajo será dual:



- Para cada laboratorio existirán 2 carpetas:
  - El directorio **source/common**: contendrá los **ficheros fuente (VHDL)** correspondientes a una biblioteca de utilidades y una colección de componentes reutilizables que podrán ser usados en cualquier proyecto.
  - Cada directorio **source/labx**: contendrá los **ficheros fuente (VHDL y XDC)** de uso exclusivo del laboratorio correspondiente.
  - Cada directorio **projects/labx**: contendrá el **fichero de proyecto (XPR)** del laboratorio correspondiente y otros ficheros/directorios auxiliares creados por Vivado.

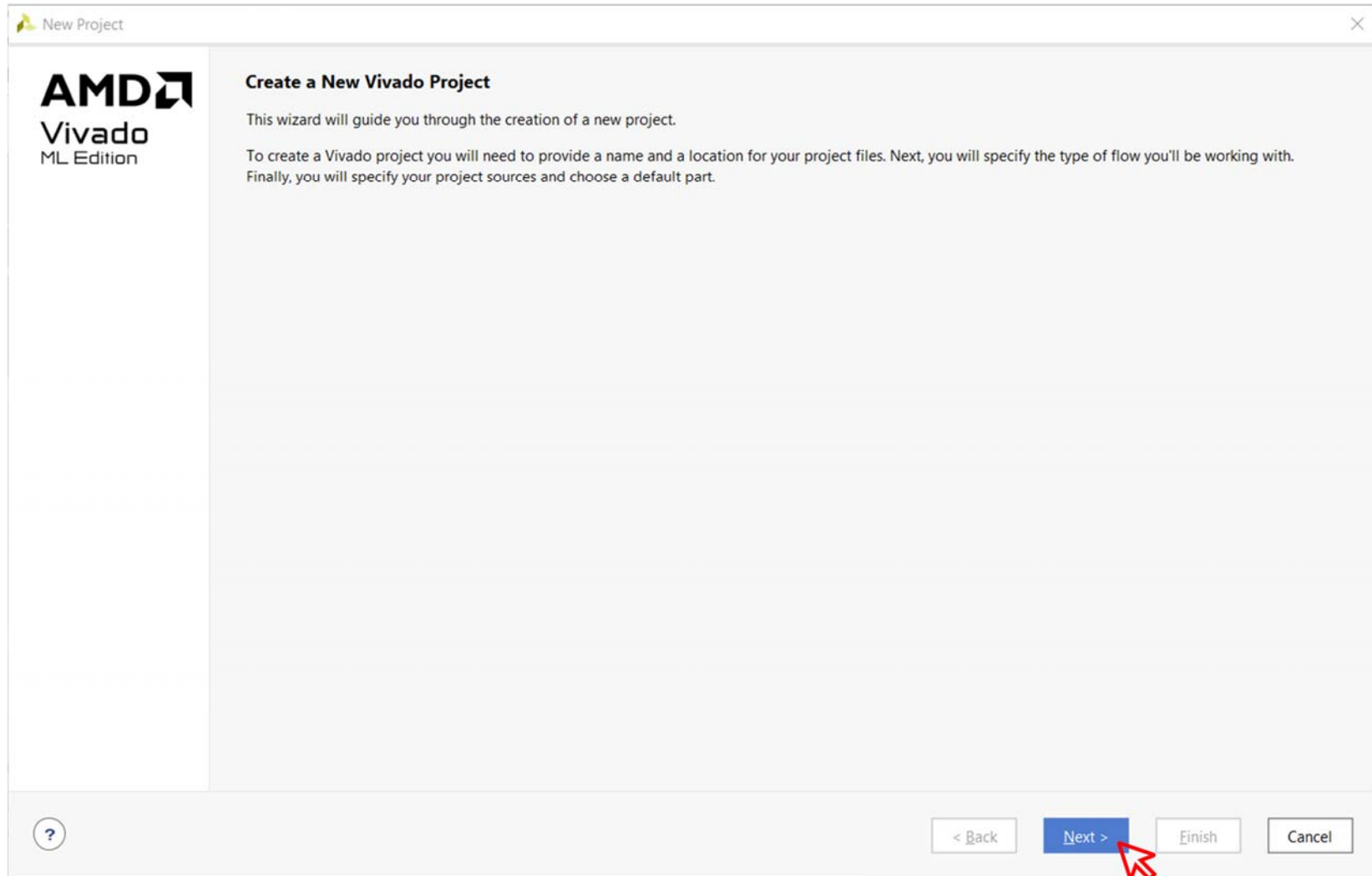
# Síntesis y prototipado

## creación del proyecto (i)



# Síntesis y prototipado

## creación del proyecto (ii)



# Síntesis y prototipado

## creación del proyecto (iii)



New Project

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

Project name:  ①

Project location:  ②

☒ Create project subdirectory

Project will be created at: C:/Users/user/Documents/DAS/labs/projects/lab1

③

Seleccionar el directorio del proyecto

# Síntesis y prototipado

## creación del proyecto (iv)



New Project

**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

☐ Project is an extensible Vitis platform

☐ **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 Project File.

☐ **Example Project**  
Create a new Vivado project from a predefined template.

? < Back **Next >** Finish Cancel

Seleccionar el tipo de proyecto

# Síntesis y prototipado

## creación del proyecto (v)



New Project

**Default Part**  
Choose a default Xilinx part or board for your project.

Parts | **Boards** ①

Reset All Filters

Vendor: All Name: All Board Rev: Latest

Search: Q

| Display Name                                                        | Preview | Status    | Vendor          | File Version | Part             | I/O Pin Count | Board Rev | Available IOBs | LUT Elements |
|---------------------------------------------------------------------|---------|-----------|-----------------|--------------|------------------|---------------|-----------|----------------|--------------|
| <b>Basys3</b>                                                       |         | Installed | digilentinc.com | 1.2          | xc7a35tcbg236-1  | 236           | C.0       | 106            | 20800        |
| Artix-7 AC701 Evaluation Platform<br>Add Companion Card Connections |         | Installed | xilinx.com      | 1.4          | xc7a200tcbg676-2 | 676           | 1.1       | 400            | 134600       |
| Zynq 7000 ZC702 Evaluation Board<br>Add Companion Card Connections  |         | Installed | xilinx.com      | 1.4          | xc7z020clg484-1  | 484           | 1.0       | 200            | 53200        |
| Zynq 7000 ZC706 Evaluation Board<br>Add Companion Card Connections  |         | Installed | xilinx.com      | 1.4          | xc7z045ffg900-2  | 900           | 1.1       | 362            | 218600       |

Refresh

②

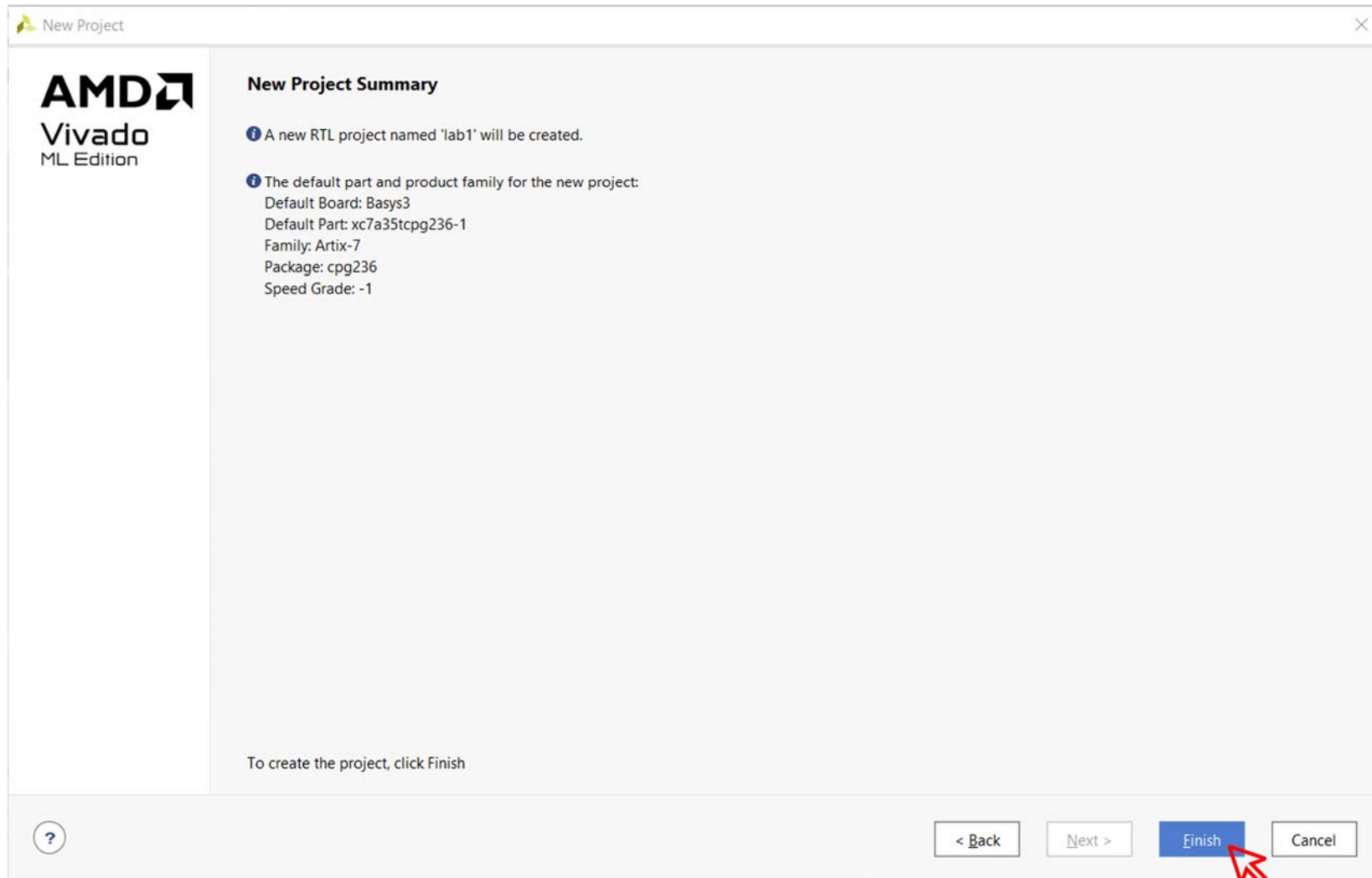
③

< Back Next > Finish Cancel

Seleccionar la placa de prototipado a usar

# Síntesis y prototipado

## creación del proyecto (vi)





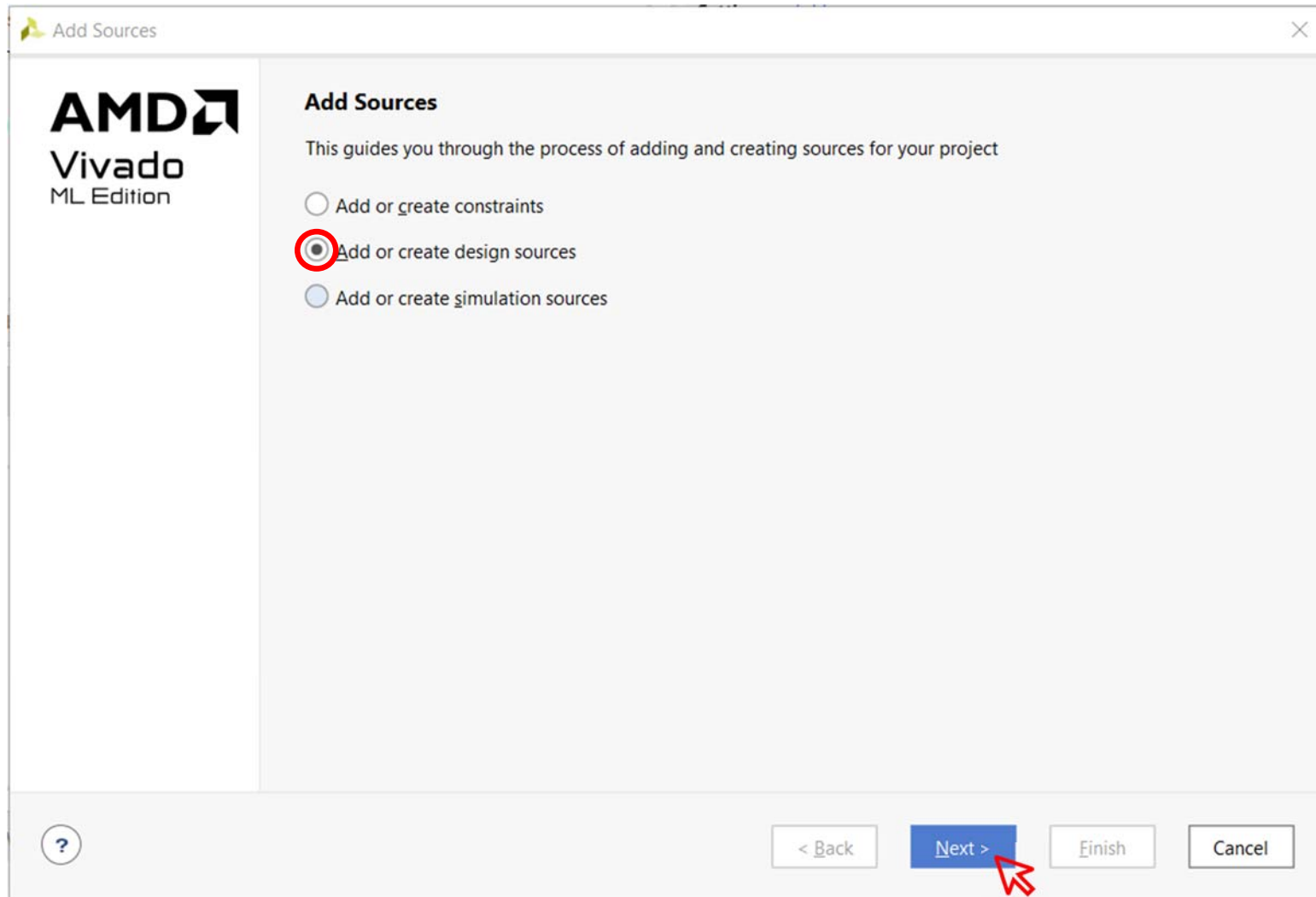
# Síntesis y prototipado

## adición de ficheros VHDL (i)

[illegible]

# Síntesis y prototipado

## adición de ficheros VHDL (ii)



# Síntesis y prototipado

## adición de ficheros VHDL (iii)



 Add Sources

### Add or Create Design Sources

Specify HDL, netlist, Block Design, and IP files, or directories containing those file types to add to your project. Create a new source file on disk and add it to your project.

+
-
↑
↓

Use Add Files, Add Directories or Create File buttons below

Add Files
Add Directories
Create File

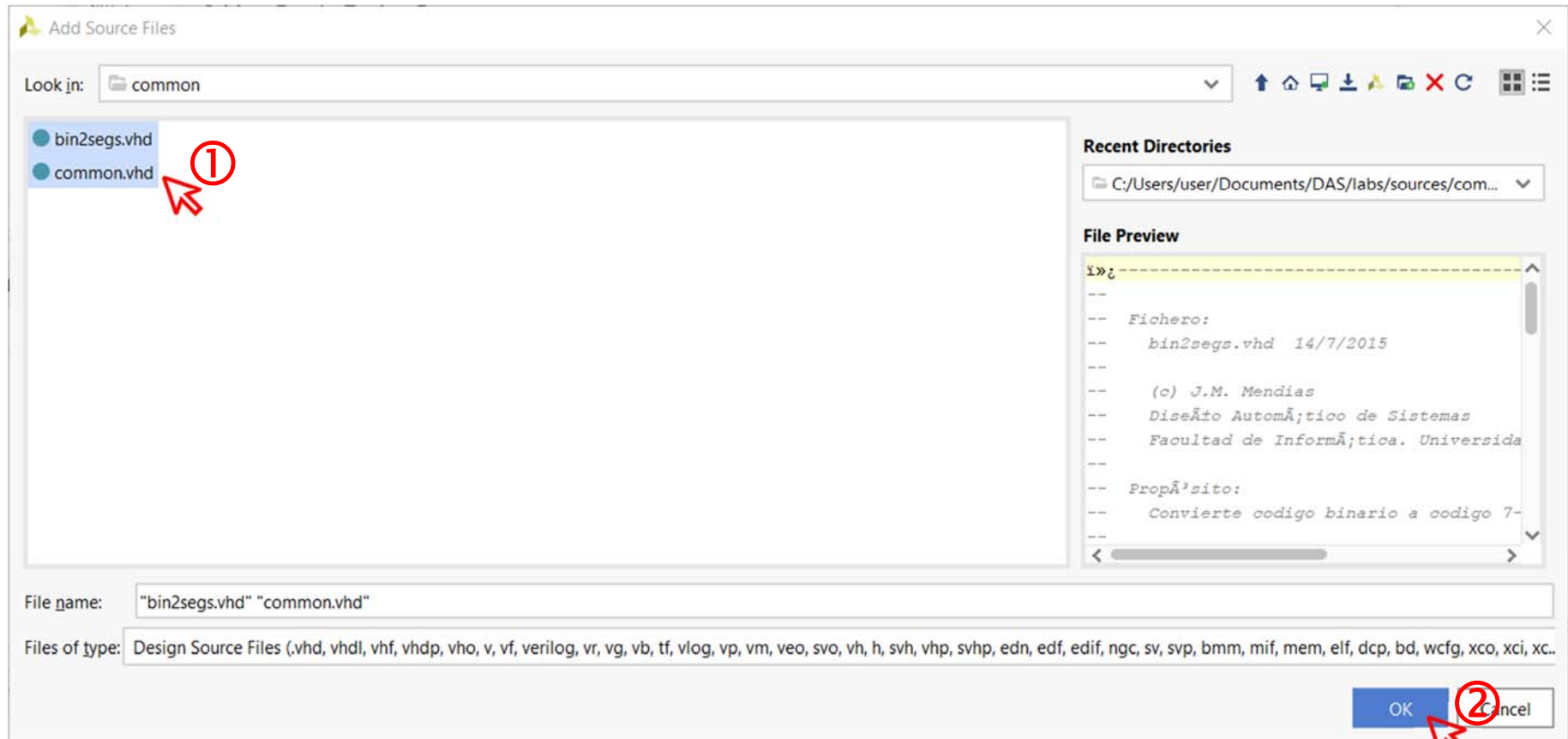
☐ Scan and add RTL include files into project  
☐ Copy sources into project  
☒ Add sources from subdirectories

?

< Back
Next >
Finish
Cancel

# Síntesis y prototipado

## adición de ficheros VHDL (iv)



# Síntesis y prototipado

## adición de ficheros VHDL (v)



Add Sources

**Add or Create Design Sources**

Specify HDL, netlist, Block Design, and IP files, or directories containing those file types to add to your project. Create a new source file on disk and add it to your project.

|   | Index | Name         | Library        | Location                                        |
|---|-------|--------------|----------------|-------------------------------------------------|
| ● | 1     | bin2segs.vhd | xil_defaultlib | C:/Users/user/Documents/DAS/labs/sources/common |
| ● | 2     | common.vhd   | xil_defaultlib | C:/Users/user/Documents/DAS/labs/sources/common |

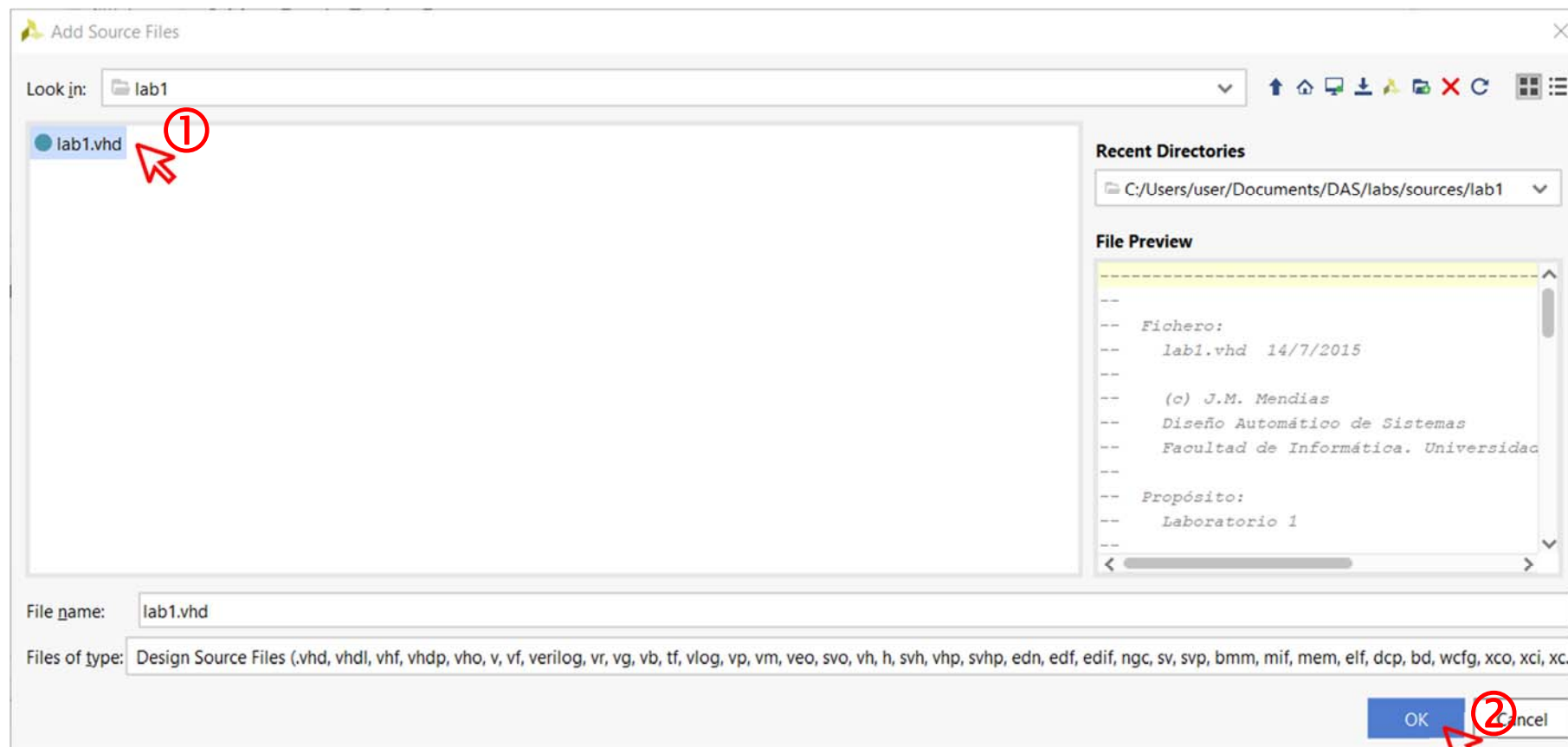
Add Files
Add Directories
Create File

☐ Scan and add RTL include files into project  
☐ Copy sources into project  
☒ Add sources from subdirectories

?
< Back
Next >
Finish
Cancel

# Síntesis y prototipado

## adición de ficheros VHDL (vi)



# Síntesis y prototipado

## adición de ficheros VHDL (vii)



Add Sources

**Add or Create Design Sources**

Specify HDL, netlist, Block Design, and IP files, or directories containing those file types to add to your project. Create a new source file on disk and add it to your project.

|   | Index | Name         | Library        | Location                                        |
|---|-------|--------------|----------------|-------------------------------------------------|
| ● | 1     | bin2segs.vhd | xil_defaultlib | C:/Users/user/Documents/DAS/labs/sources/common |
| ● | 2     | common.vhd   | xil_defaultlib | C:/Users/user/Documents/DAS/labs/sources/common |
| ● | 3     | lab1.vhd     | xil_defaultlib | C:/Users/user/Documents/DAS/labs/sources/lab1   |

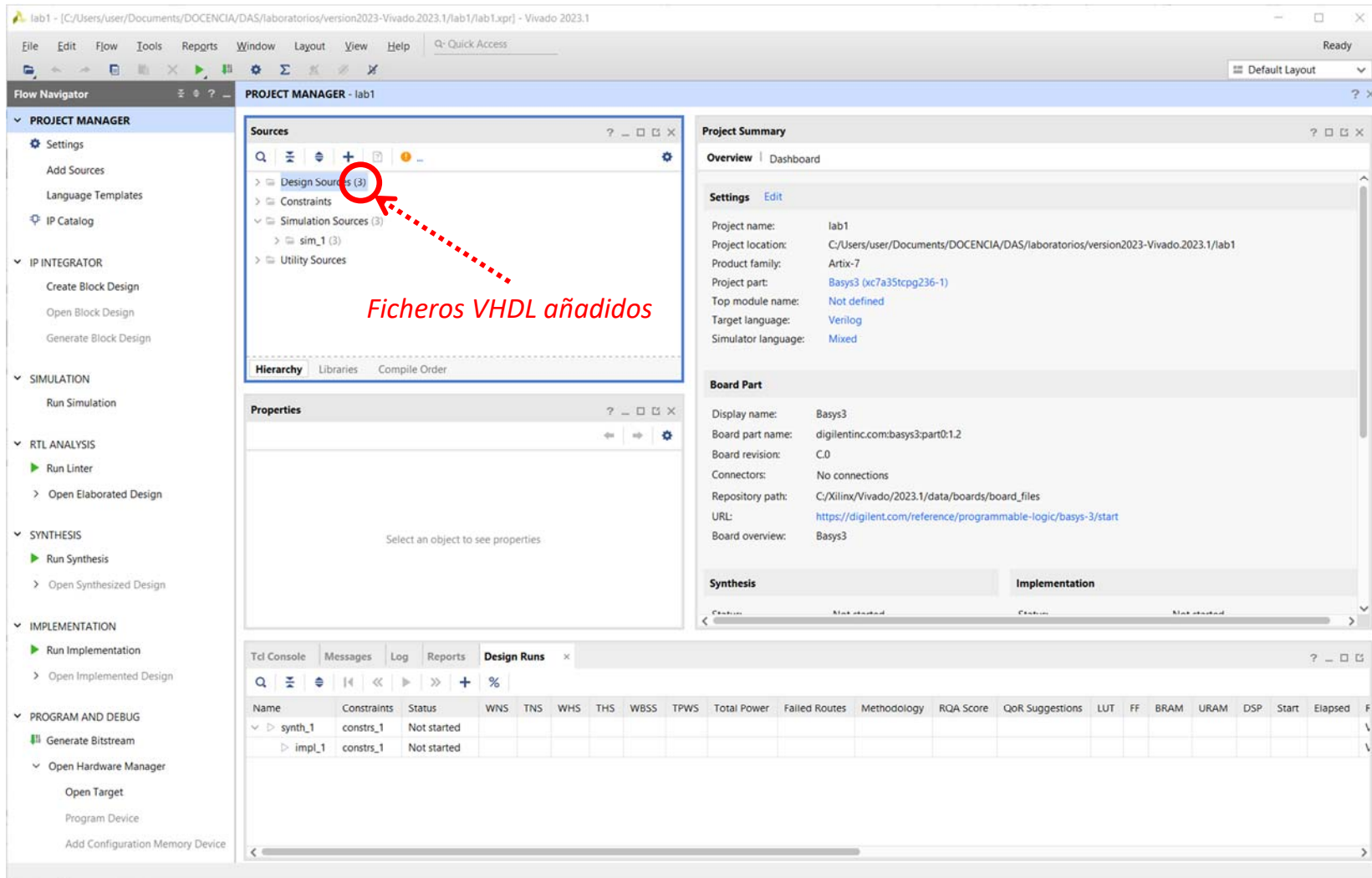
Add Files
 Add Directories
 Create File

☐ Scan and add RTL include files into project  
☒ Copy sources into project  
☒ Add sources from subdirectories

?
 < Back
 Next >
 Finish
 Cancel

# Síntesis y prototipado

## adición de ficheros VHDL (viii)

[illegible]



# Síntesis y prototipado

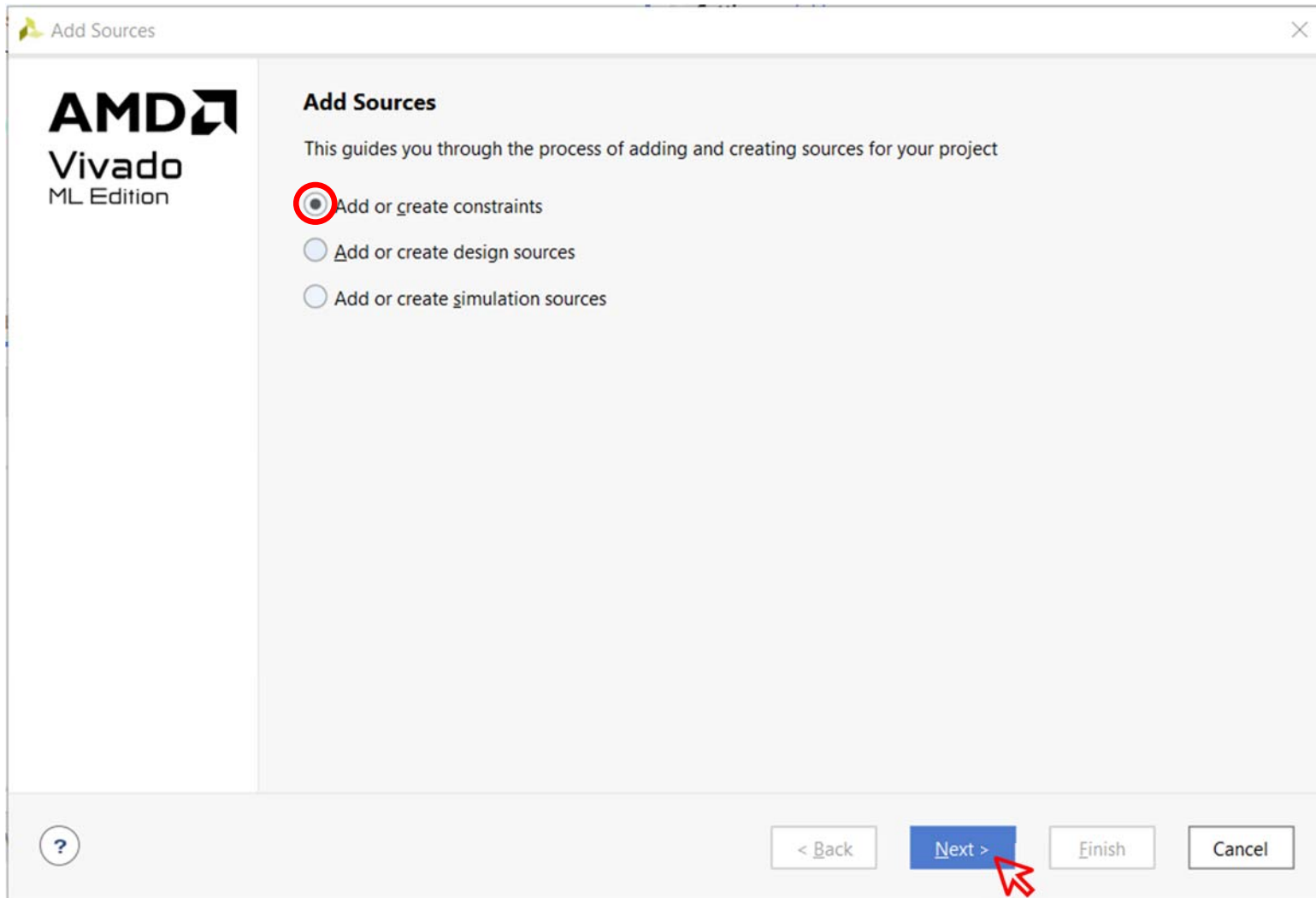
## adición del fichero XDC de ligaduras (i)

The screenshot displays the Vivado IDE interface with the following components:

- Flow Navigator (Left Panel):** Contains a tree view with categories like PROJECT MANAGER, IP INTEGRATOR, SIMULATION, RTL ANALYSIS, SYNTHESIS, IMPLEMENTATION, and PROGRAM AND DEBUG. The 'Add Sources' option under PROJECT MANAGER is highlighted with a red box and a red arrow.
- Project Manager - lab1 (Top Center Panel):** Shows the project hierarchy with sections for Sources, Libraries, and Compile Order. The Sources section is expanded, showing Design Sources (3), Constraints, Simulation Sources (3), and Utility Sources.
- Sources (Bottom Center Panel):** A panel for managing design sources, currently showing a list of sources.
- Properties (Bottom Center Panel):** A panel for viewing the properties of a selected object, currently showing 'Select an object to see properties'.
- Project Summary (Right Panel):** Provides an overview of the project, including settings like Project name, Project location, Product family, Project part, Top module name, Target language, and Simulator language.
- Design Runs (Bottom Panel):** A table showing the status of design runs. The table has columns for Name, Constraints, Status, WNS, TNS, WHS, THS, WBSS, TPWS, Total Power, Failed Routes, Methodology, RQA Score, QoR Suggestions, LUT, FF, BRAM, URAM, DSP, Start, and Elapsed. The table shows two runs: 'synth\_1' and 'impl\_1', both with a status of 'Not started'.

# Síntesis y prototipado

## adición del fichero XDC de ligaduras (ii)



# Síntesis y prototipado

## adición del fichero XDC de ligaduras (iii)



Add Sources

### Add or Create Constraints

Specify or create constraint files for physical and timing constraint to add to your project.

Specify constraint set: constrs\_1 (active)

+
-
↑
↓

Use Add Files or Create File buttons below

Add Files
Create File

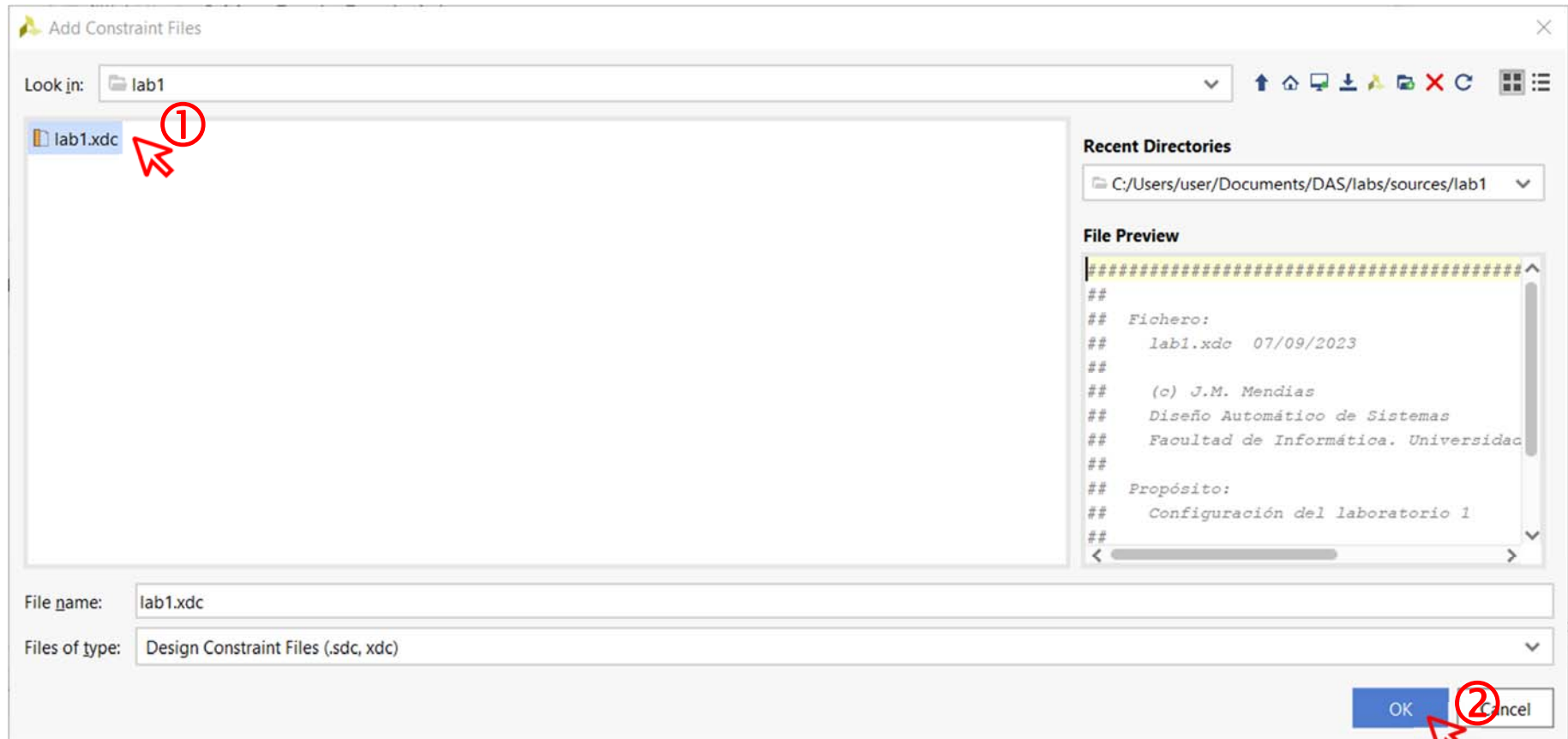
☒ Copy constraints files into project

?

< Back
Next >
Finish
Cancel

# Síntesis y prototipado

## adición del fichero XDC de ligaduras (iv)



# Síntesis y prototipado

## adición del fichero XDC de ligaduras (v)



**Add Sources**

**Add or Create Constraints**

Specify or create constraint files for physical and timing constraint to add to your project.

Specify constraint set: constrs\_1 (active)

| Constraint File | Location                     |
|-----------------|------------------------------|
| lab1.xdc        | C:\Users\user\Documents\lab1 |

☐ Copy constraints files into project



# Síntesis y prototipado

## adición del fichero XDC de ligaduras (vi)

[illegible]

# Síntesis y prototipado

## implementación



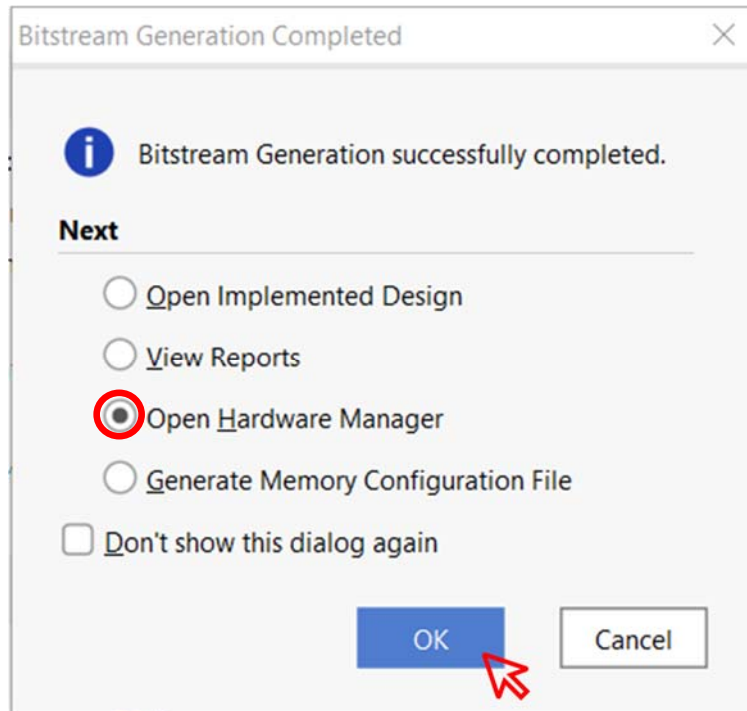
The screenshot displays the Vivado ML Edition software interface. The top menu bar includes File, Edit, Flow, Tools, Reports, Window, Layout, View, and Help. The left sidebar shows the Flow Navigator with various project management and synthesis options. The main workspace is divided into several panels:

- PROJECT MANAGER - lab1**: This panel shows the project's structure, including Design Sources (3), Constraints (1), Simulation Sources (3), and Utility Sources. The 'Hierarchy' tab is selected, showing a tree view of the project components.
- Project Summary**: This panel provides an overview of the project settings, including the project name (lab1), location, product family (Artix-7), project part (Basys3), top module name (bin2segs), target language (Verilog), and simulator language (Mixed).
- Board Part**: This panel shows the board part details, including the display name (Basys3), board part name (digilentinc.com:basys3:part0:1.2), board revision (C.0), connectors (No connections), repository path, URL, and board overview (Basys3).
- Design Runs**: This panel shows the status of the design runs. The 'Design Runs' tab is selected, displaying a table with columns for Name, Constraints, Status, WNS, TNS, WHS, THS, WBSS, TPWS, Total Power, Failed Routes, Methodology, RQA Score, QoR Suggestions, LUT, FF, BRAM, URAM, DSP, Start, Elapsed, and F. The table shows two runs: 'synth\_1' (constraints\_1, Not started) and 'impl\_1' (constraints\_1, Not started).

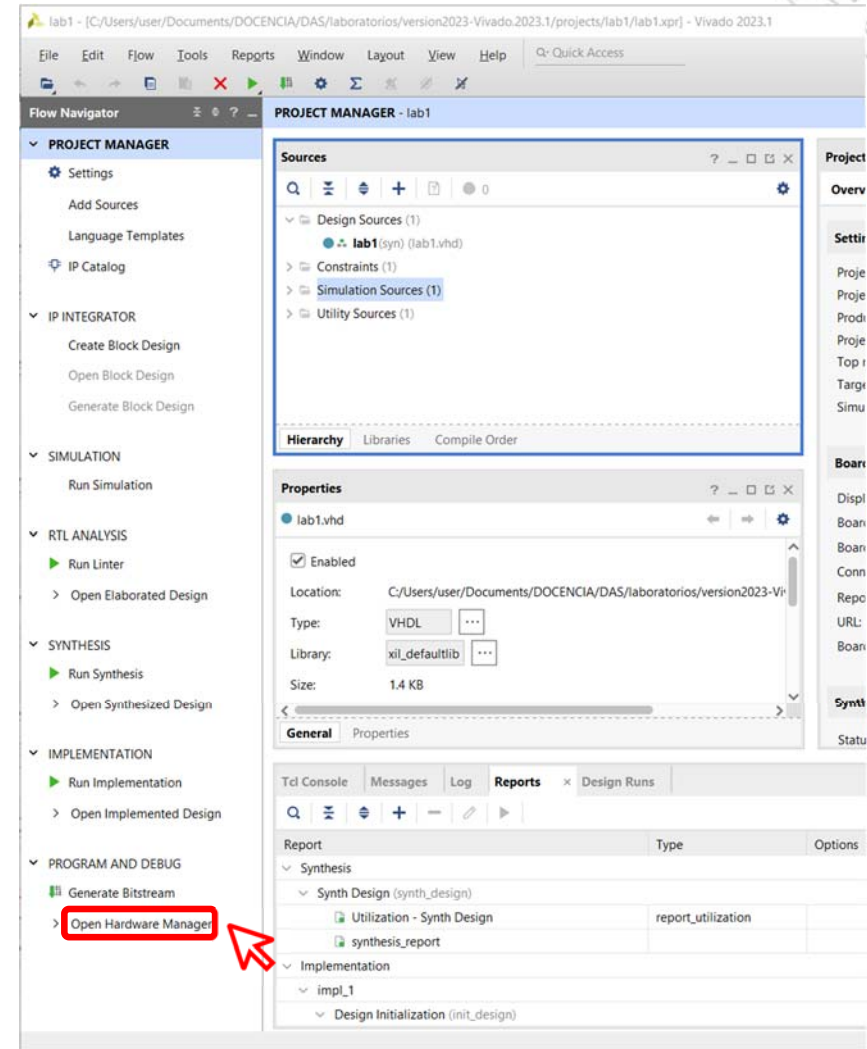
In the bottom left corner of the Project Manager panel, the 'Generate Bitstream' option is highlighted with a red box and a red arrow pointing to it.

# Síntesis y prototipado

## volcado del fichero BIT de configuración (i)



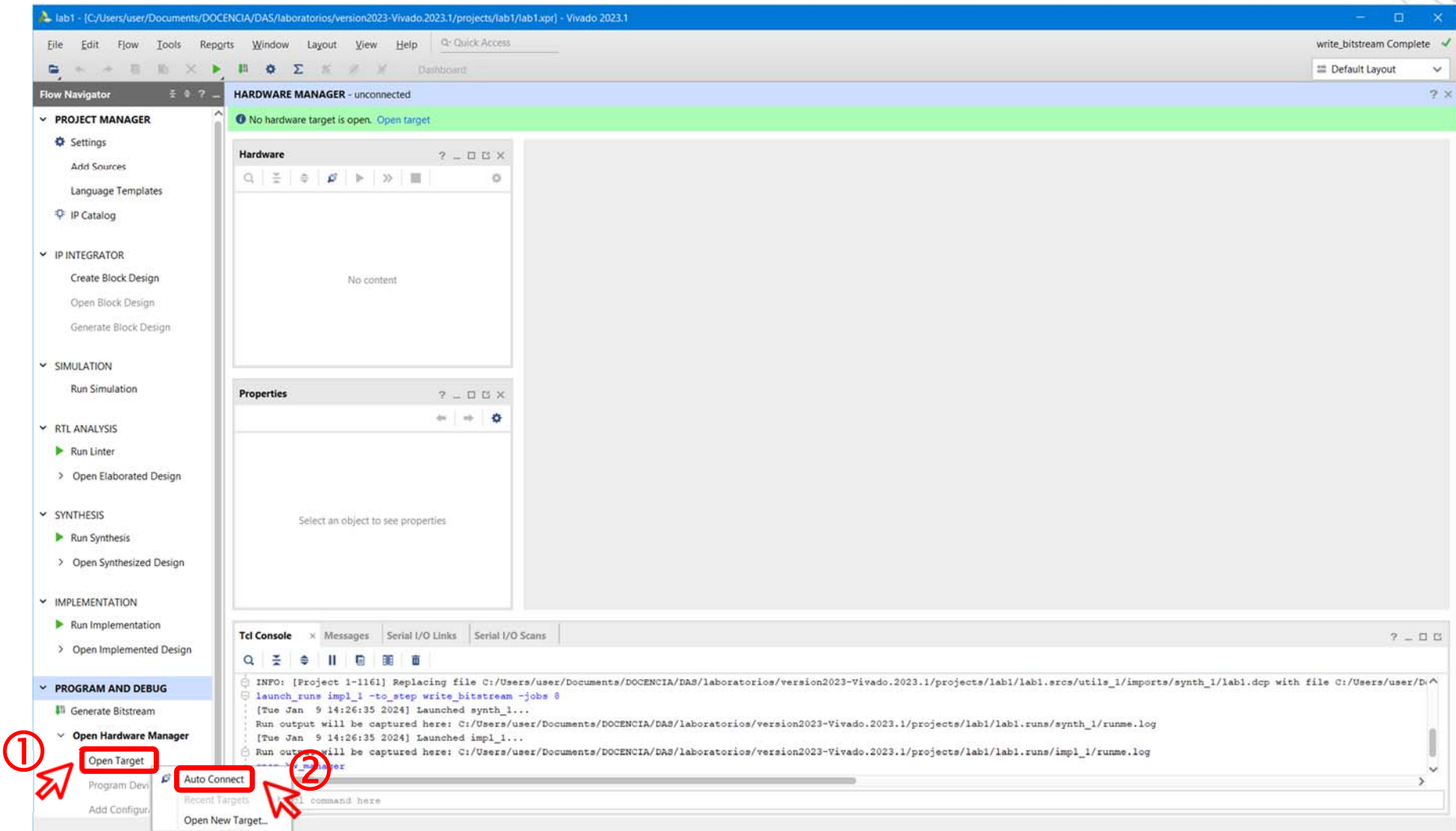
Opción 1



Opción 2

# Síntesis y prototipado

## volcado del fichero BIT de configuración (ii)



# Síntesis y prototipado

## volcado del fichero BIT de configuración (iii)



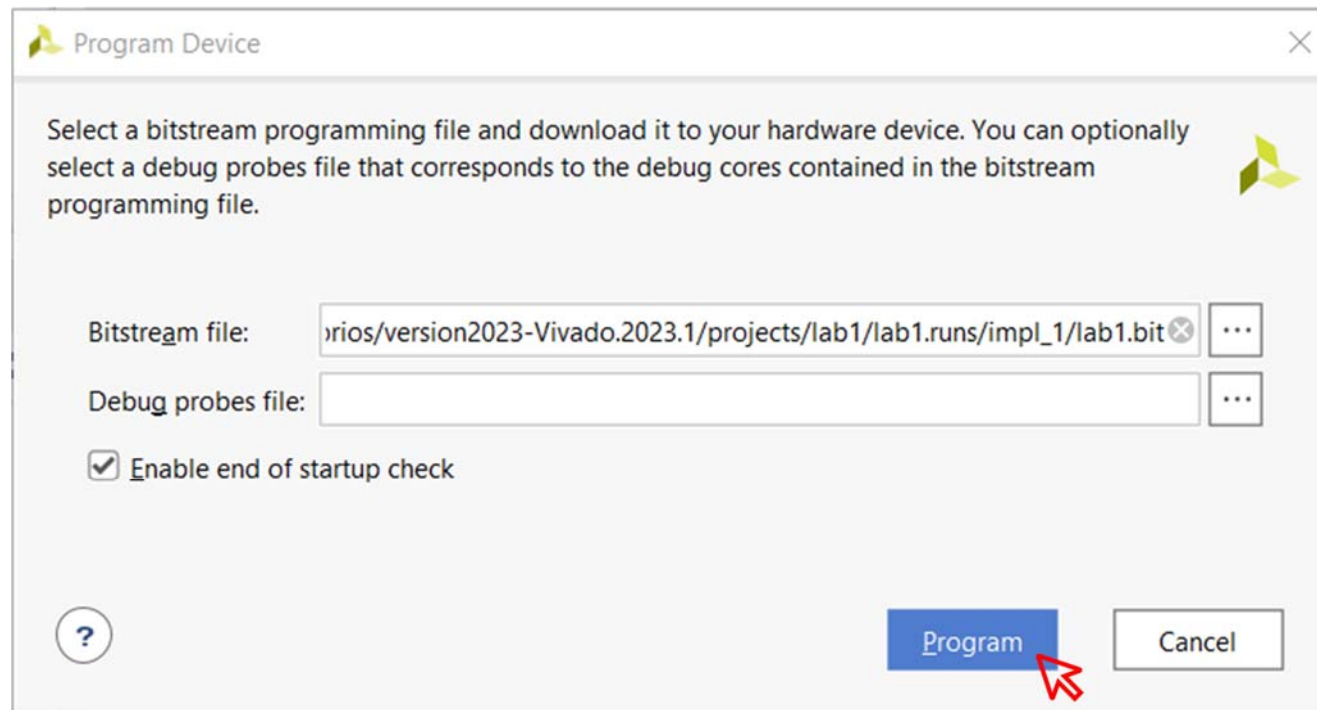
The screenshot shows the Vivado 2023.1 IDE interface. The left sidebar contains the 'PROJECT MANAGER' and 'PROGRAM AND DEBUG' sections. The 'PROGRAM AND DEBUG' section is expanded, showing 'Generate Bitstream' and 'Open Hardware Manager'. The 'Open Hardware Manager' option is selected, and a red box highlights it with a red arrow and the number 1. Below it, the 'xc7a35t\_0' device is also highlighted with a red box and a red arrow and the number 2. The main window displays the 'HARDWARE MANAGER' for the target 'localhost/xilinx\_tcf/Digilent/210183A6F765A'. It shows a table of hardware components: 'localhost (1)' (Connected), 'xilinx\_tcf/Digilent/210183A6F765A' (Open), and 'xc7a35t\_0 (1)' (Not programmed). The 'Properties' panel is empty, showing 'Select an object to see properties'. The bottom panel shows the 'Tcl Console' with the following output:

```

open_hw_target
INFO: [Labtools141 44-466] Opening hw_target localhost:3121/xilinx_tcf/Digilent/210183A6F765A
set_property PROGRAM.FILE [C:/Users/user/Documents/DOCENCIA/DAS/laboratorios/version2023-Vivado.2023.1/projects/lab1/lab1.runs/impl_1/lab1.bit] [get_hw_devices xc7a35t_0]
current_hw_device [get_hw_devices xc7a35t_0]
refresh_hw_device -update_hw_probes false [lindex [get_hw_devices xc7a35t_0] 0]
INFO: [Labtools 27-1435] Device xc7a35t (JTAG device index = 0) is not programmed (DONE status = 0).
  
```

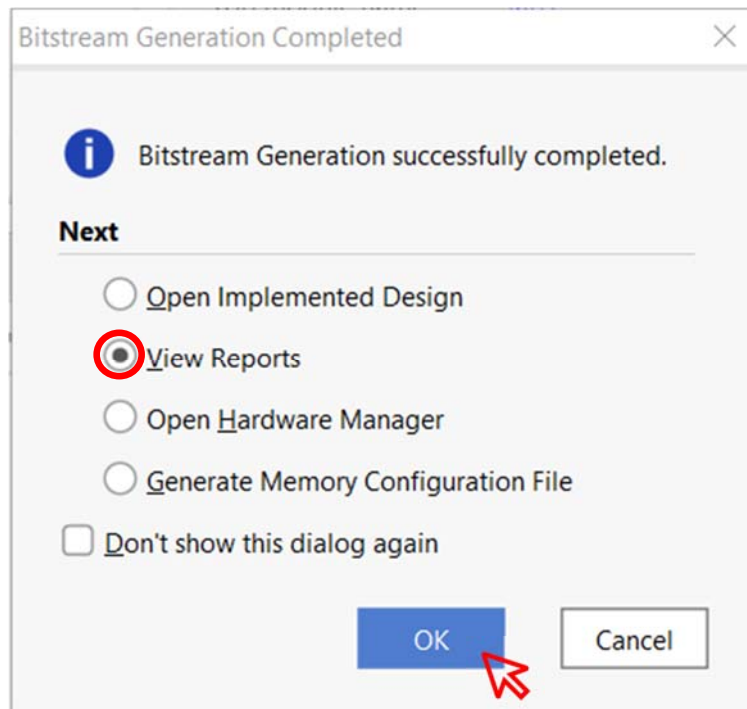
# Síntesis y prototipado

## volcado del fichero BIT de configuración (iv)

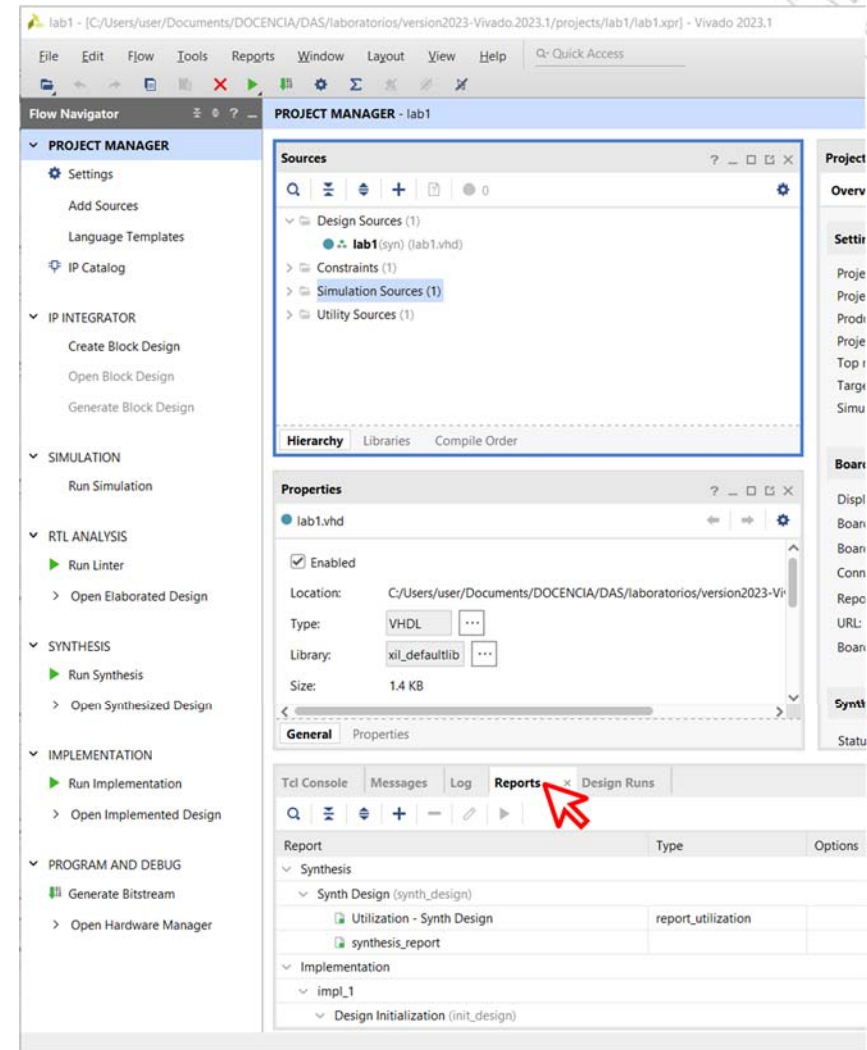


# Síntesis y prototipado

## visualización de reports (i)



Opción 1



Opción 2

# Síntesis y prototipado

## visualización de reports (ii)



The screenshot shows the Vivado ML Edition interface. The left sidebar contains the Project Manager, IP Integrator, Simulation, RTL Analysis, Synthesis, and Implementation sections. The main window is divided into several panes. The top pane shows the Project Manager for 'lab1'. The bottom pane shows the Reports window, which is currently displaying the 'Utilization - Synth Design' report. The report is titled 'Utilization - Synth Design - synth\_1' and shows the path 'C:/Users/user/Documents/DOCENCIA/DAS/laboratorios/version2023-Vivado.2023.1/projects/lab1/lab1.runs/synth\_1/lab1\_utilization\_synth.rpt'. The report content includes a table of contents with sections like '1. Slice Logic', '1.1 Summary of Registers by Type', '2. Memory', '3. DSP', '4. IO and GT Specific', '5. Clocking', '6. Specific Feature', '7. Primitives', '8. Black Boxes', '9. Instantiated Netlists', and '1. Slice Logic'.

Two red arrows point to the 'Reports' tab in the bottom pane, labeled with circled numbers 1 and 2. Arrow 1 points to the 'Reports' tab, and Arrow 2 points to the 'Utilization - Synth Design' report entry in the list.

| Report                              | Type               | Options | Modified        | Size    |
|-------------------------------------|--------------------|---------|-----------------|---------|
| Synthesis                           |                    |         |                 |         |
| Synth Design (synth_design)         |                    |         |                 |         |
| Utilization - Synth Design          | report_utilization |         | 1/9/24, 2:28 PM | 7.9 KB  |
| synthesis_report                    |                    |         | 1/9/24, 2:28 PM | 16.7 KB |
| Implementation                      |                    |         |                 |         |
| impl_1                              |                    |         |                 |         |
| Design Initialization (init_design) |                    |         |                 |         |

# Análisis nivel RT



J.M. Méndias  
2024

tema 3:  
El entorno de diseño Vivado ML Edition

lab1 - [C:/Users/user/Documents/DOCENCIA/DAS/laboratorios/version2023-Vivado.2023.1/projects/lab1/lab1.xpr] - Vivado 2023.1

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

write\_bitstream Complete ✓  
Default Layout

Flow Navigator

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

- Run Linter
- Open Elaborated Design
- Report Methodology
- Report DRC
- Report Noise
- Schematic
- Open Dataflow Design

SYNTHESIS

- Run Synthesis
- Open Synthesized Design

IMPLEMENTATION

- Run Implementation
- Open Implemented Design

PROGRAM AND DEBUG

ELABORATED DESIGN - xc7a35tcbg236-1

Sources Netlist

- lab1
  - Nets (75)
  - Leaf Cells (6)

Properties

Select an object to see properties

Project Summary Schematic

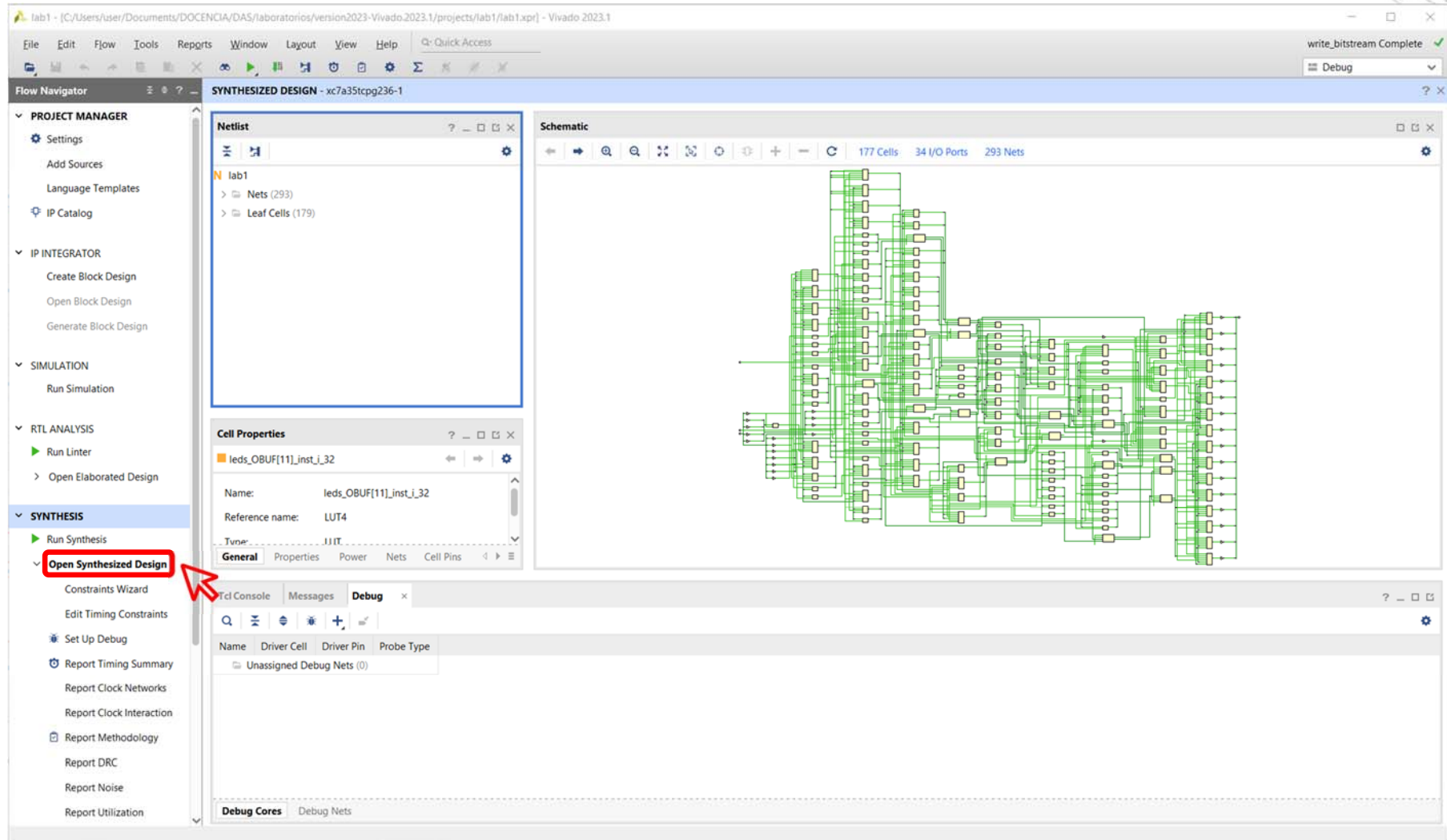
5 Cells 34 I/O Ports 75 Nets

Diagram showing RTL components: ARG\_i, ARG\_i\_0, ARG\_i\_1, result0\_i, RTL\_ADD, RTL\_SUB, RTL\_MINUS, RTL\_MUX, and result\_i. Inputs include btnL, btnR, and switches[15:0]. Outputs include leds[15:0].

Tcl Console Messages Log Reports Design Runs

| Name    | Constraints | Status                    | WNS | TNS | WHS | THS | WBSS | TPWS | Total Power | Failed Routes | Methodology | RQA Score | QoR Suggestions | LUT | FF | BRAM | URAM | DSP | Start           | Elapsed  | Run S |
|---------|-------------|---------------------------|-----|-----|-----|-----|------|------|-------------|---------------|-------------|-----------|-----------------|-----|----|------|------|-----|-----------------|----------|-------|
| synth_1 | constrs_1   | synth_design Complete!    |     |     |     |     |      |      |             |               |             |           |                 | 99  | 0  | 0    | 0    | 0   | 1/9/24, 2:26 PM | 00:01:32 | Vivad |
| impl_1  | constrs_1   | write_bitstream Complete! | NA  | NA  | NA  | NA  |      | NA   | 33.654      | 0             |             |           |                 | 99  | 0  | 0    | 0    | 0   | 1/9/24, 2:28 PM | 00:02:17 | Vivad |

# Análisis nivel lógico



# Análisis nivel físico



lab1 - [C:/Users/user/Documents/DOCENCIA/DAS/laboratorios/version2023-Vivado2023.1/projects/lab1/lab1.xpr] - Vivado 2023.1

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

write\_bitstream Complete ✓  
Default Layout

Flow Navigator

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

- Run Linter
- Open Elaborated Design

SYNTHESIS

- Run Synthesis
- Open Synthesized Design

IMPLEMENTATION

- Run Implementation
- Open Implemented Design**
- Constraints Wizard
- Edit Timing Constraints
- Report Timing Summary
- Report Clock Networks
- Report Clock Interaction
- Report Methodology

IMPLEMENTED DESIGN - xc7a35tcbg236-1

Sources Netlist

- lab1
  - Nets (251)
  - Leaf Cells (179)

Properties

Select an object to see properties

Project Summary Device

Tcl Console Messages Log Reports Design Runs Power Timing

| Name    | Constraints | Status                    | WNS | TNS | WHS | THS | WBSS | TPWS | Total Power | Failed Routes | Methodology | RQA Score | QoR Suggestions | LUT | FF | BRAM | URAM | DSP | Start           | Elapsed  | Run S |
|---------|-------------|---------------------------|-----|-----|-----|-----|------|------|-------------|---------------|-------------|-----------|-----------------|-----|----|------|------|-----|-----------------|----------|-------|
| synth_1 | constrs_1   | synth_design Complete!    |     |     |     |     |      |      |             |               |             |           |                 | 99  | 0  | 0    | 0    | 0   | 1/9/24, 2:26 PM | 00:01:32 | Vivad |
| impl_1  | constrs_1   | write_bitstream Complete! | NA  | NA  | NA  | NA  |      | NA   | 33.654      | 0             |             |           |                 | 99  | 0  | 0    | 0    | 0   | 1/9/24, 2:28 PM | 00:02:17 | Vivad |

Save the constraints for the current design

# Validación por simulación

## adición del testbench VHDL (i)



The screenshot shows the Vivado 2023.1 interface with the Project Manager and Project Summary panels. The Project Manager panel on the left shows the 'Add Sources' button highlighted with a red box and a mouse cursor. The Project Summary panel on the right shows the project details for 'lab1'.

**Project Manager - lab1**

- Settings
- Add Sources** (highlighted)
- Language Templates
- IP Catalog
- IP INTEGRATOR
  - Create Block Design
  - Open Block Design
  - Generate Block Design
- SIMULATION
  - Run Simulation
- RTL ANALYSIS
  - Run Linter
  - Open Elaborated Design
- SYNTHESIS
  - Run Synthesis
  - Open Synthesized Design
- IMPLEMENTATION
  - Run Implementation
  - Open Implemented Design
- PROGRAM AND DEBUG
  - Generate Bitstream
  - Open Hardware Manager

**Project Summary**

**Overview | Dashboard**

**Settings | Edit**

Project name: lab1  
 Project location: C:/Users/user/Documents/DOCENCIA/DAS/laboratorios/version2023-Vivado.2023.1/projects/lab1  
 Product family: Artix-7  
 Project part: Basys3 (xc7a35tcbg236-1)  
 Top module name: lab1  
 Target language: Verilog  
 Simulator language: Mixed

**Board Part**

Display name: Basys3  
 Board part name: digilentinc.com:basys3:part0:1.2  
 Board revision: C.0  
 Connectors: No connections  
 Repository path: C:/Xilinx/Vivado/2023.1/data/boards/board\_files  
 URL: <https://digilent.com/reference/programmable-logic/basys-3/start>  
 Board overview: Basys3

**Synthesis | Implementation | Summary | Route Status**

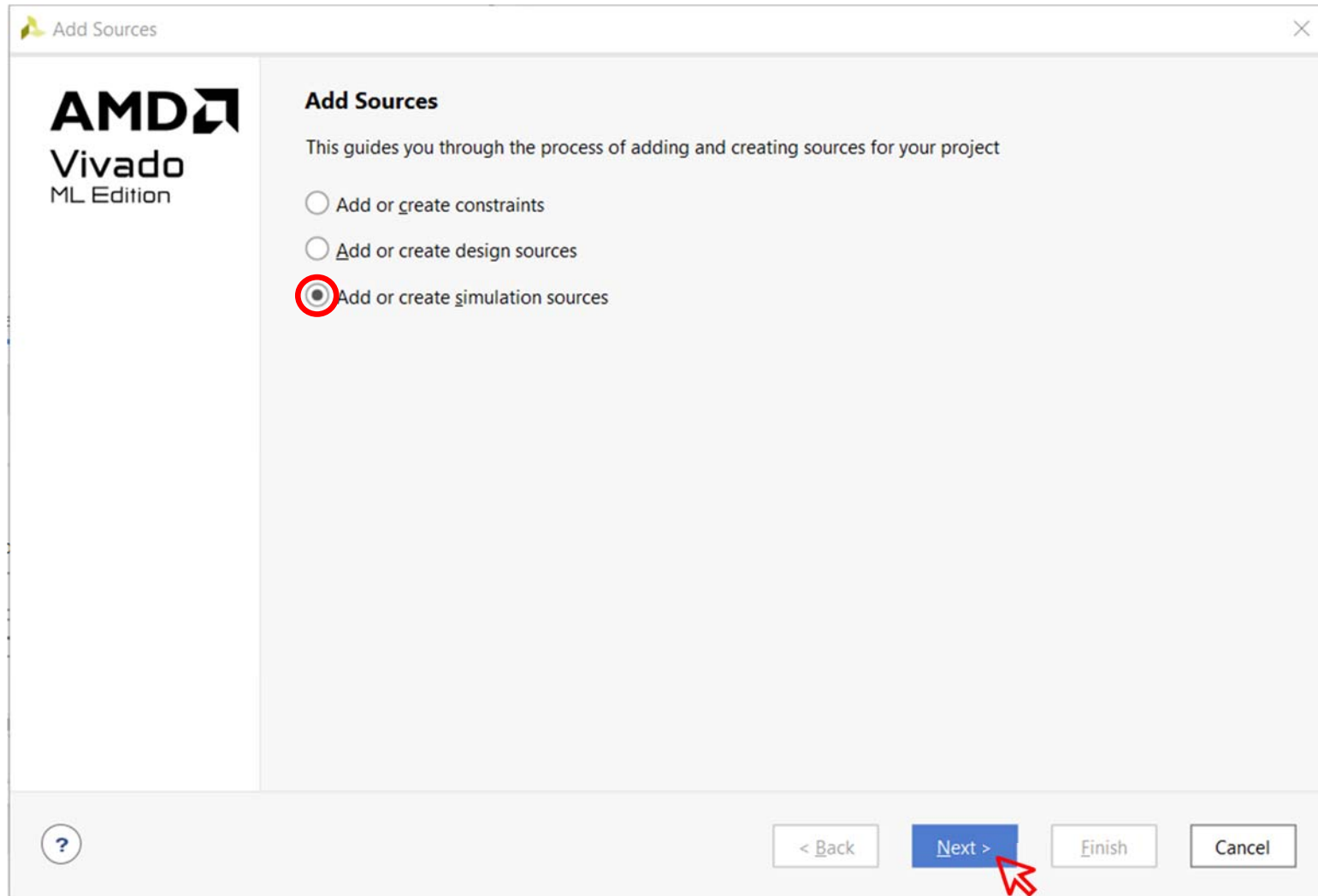
Status: ✓ Complete

**Tcl Console | Messages | Log | Reports | Design Runs**

| Report                                | Type               | Options | Modified        | Size    |
|---------------------------------------|--------------------|---------|-----------------|---------|
| ▼ Synthesis                           |                    |         |                 |         |
| ▼ Synth Design (synth_design)         |                    |         |                 |         |
| Utilization - Synth Design            | report_utilization |         | 1/9/24, 2:28 PM | 7.9 KB  |
| synthesis_report                      |                    |         | 1/9/24, 2:28 PM | 16.7 KB |
| ▼ Implementation                      |                    |         |                 |         |
| ▼ impl_1                              |                    |         |                 |         |
| ▼ Design Initialization (init_design) |                    |         |                 |         |

# Validación por simulación

## adición del testbench VHDL (ii)



# Validación por simulación

## adición del testbench VHDL (iii)



Add Sources

### Add or Create Simulation Sources

Specify simulation specific HDL files, or directories containing HDL files, to add to your project. Create a new source file on disk and add it to your project.

Specify simulation set: sim\_1

+
-
↑
↓

Use Add Files, Add Directories or Create File buttons below

Add Files
Add Directories
Create File

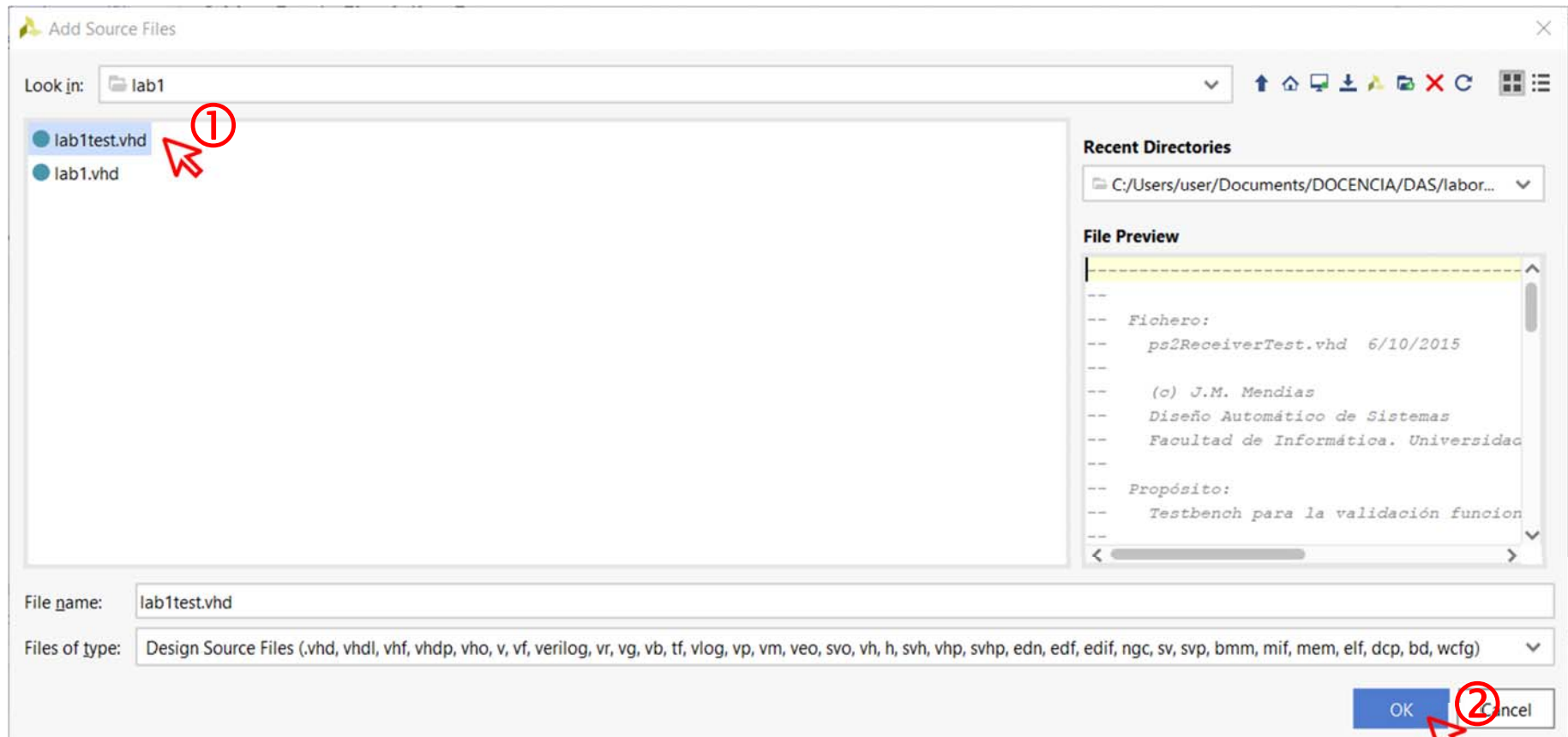
☐ Scan and add RTL include files into project  
☐ Copy sources into project  
☒ Add sources from subdirectories  
☒ Include all design sources for simulation

?

< Back
Next >
Finish
Cancel

# Validación por simulación

## adición del testbench VHDL (iv)



# Validación por simulación

## adición del testbench VHDL (v)



Add Sources

### Add or Create Simulation Sources

Specify simulation specific HDL files, or directories containing HDL files, to add to your project. Create a new source file on disk and add it to your project.

Specify simulation set: sim\_1

|  | Index | Name         | Library        | Location                                                                            |
|--|-------|--------------|----------------|-------------------------------------------------------------------------------------|
|  | 1     | lab1test.vhd | xil_defaultlib | C:/Users/user/Documents/DOCENCIA/DAS/laboratorios/version2023-Vivado.2023.1/sources |

Add Files
Add Directories
Create File

☐ Scan and add RTL include files into project  
☒ Copy sources into project  
☒ Add sources from subdirectories  
☒ Include all design sources for simulation

?
< Back
Next >
Finish
Cancel

# Validación por simulación

## adición del testbench VHDL (vi)



The screenshot shows the Vivado 2023.1 interface with the following components:

- Flow Navigator:** Shows the project workflow from Settings to Program and Debug.
- PROJECT MANAGER - lab1:**
  - Sources:** Lists Design Sources (1), Constraints (1), Simulation Sources (1), and Utility Sources (1). A red circle highlights the Simulation Sources entry, with a red arrow pointing to the text "Testbench VHDL añadido".
  - Source File Properties:** Shows details for "lab1test.vhd", including its location, type (VHDL), library (xil\_defaultlib), and size (1.9 KB).
- Project Summary:**
  - Overview:** Displays project details such as Project name (lab1), Project location, Product family (Artix-7), Project part (Basys3), Top module name (lab1), Target language (Verilog), and Simulator language (Mixed).
  - Board Part:** Shows board information like Display name (Basys3), Board part name, Board revision, Connectors, Repository path, URL, and Board overview.
  - Synthesis and Implementation:** Both sections show a status of "Complete".
- Reports:** A table showing the results of the synthesis and implementation runs.
 

| Report                              | Type               | Options | Modified        | Size    |
|-------------------------------------|--------------------|---------|-----------------|---------|
| Synthesis                           |                    |         |                 |         |
| Synth Design (synth_design)         |                    |         |                 |         |
| Utilization - Synth Design          | report_utilization |         | 1/9/24, 2:28 PM | 7.9 KB  |
| synthesis_report                    |                    |         | 1/9/24, 2:28 PM | 16.7 KB |
| Implementation                      |                    |         |                 |         |
| impl_1                              |                    |         |                 |         |
| Design Initialization (init_design) |                    |         |                 |         |

# Validación por simulación

## selección del tipo de simulación



The screenshot displays the Vivado 2023.1 IDE interface for a project named 'lab1'. The 'Run Simulation' button in the 'SIMULATION' section is highlighted with a red circle and arrow (1). The 'Run Behavioral Simulation' option in the dropdown menu is highlighted with a red circle and arrow (2). The 'Project Summary' panel on the right shows project details, including the project name, location, product family, and target language. The 'Reports' panel at the bottom shows the status of synthesis and implementation, both marked as 'Complete'.

| Report                              | Type               | Options | Modified        | Size    |
|-------------------------------------|--------------------|---------|-----------------|---------|
| Synthesis                           |                    |         |                 |         |
| Synth Design (synth_design)         |                    |         |                 |         |
| Utilization - Synth Design          | report_utilization |         | 1/9/24, 2:28 PM | 7.9 KB  |
| synthesis_report                    |                    |         | 1/9/24, 2:28 PM | 16.7 KB |
| Implementation                      |                    |         |                 |         |
| impl_1                              |                    |         |                 |         |
| Design Initialization (init_design) |                    |         |                 |         |

# Validación por simulación

## simulación



lab1 - [C:/Users/user/Documents/DOCENCIA/DAS/laboratorios/version2023-Vivado2023.1/projects/lab1/lab1.xpr] - Vivado 2023.1

File Edit Flow Tools Reports Window Layout View Run Help Quick Access: write\_bitstream Complete

Flow Navigator: PROJECT MANAGER, IP INTEGRATOR, SIMULATION, RTL ANALYSIS, SYNTHESIS, IMPLEMENTATION, PROGRAM AND DEBUG

**Scope**

| Name     | Design Unit   | Block Type  |
|----------|---------------|-------------|
| lab1Test | lab1Test(sim) | VHDL Entity |
| uut      | lab1(syn)     | VHDL Entity |

**Objects**

| Name           | Value | Data Type |
|----------------|-------|-----------|
| switches[15:0] | 0202  | Array     |
| btnL           | 1     | Logic     |
| btnR           | 1     | Logic     |
| leds[15:0]     | 0004  | Array     |

**Waveform**

| Name   | Value |
|--------|-------|
| swi... | 0202  |
| btnL   | 1     |
| btnR   | 1     |
| led... | 0004  |

**Tcl Console**

```
run 1000 ns
stop
run 100 ms
run 500 ms
```

Sim Time: 600002 us

# Controlando Vivado

## comandos (i)



- Vivado es una **herramienta automática de especificación y síntesis**:
  - El diseñador **especifica mediante VHDL** el comportamiento funcional deseado.
  - Una colección de **algoritmos decide la implementación** resultante dirigidos por una serie de **ligaduras** impuestas por el diseñador.
  - La herramienta se controla **ejecutando comandos TCL** (*Tool Command Language*) que pueden ser lanzados **implícitamente por el IDE** o **explícitamente por el diseñador**.
- El diseñador puede optar por 2 modos de trabajo:
  - **Modo con proyecto**: Vivado gestiona los comandos TCL que ejecuta y crea una estructura de directorios para almacenar los resultados de cada fase del diseño.
    - Es el modo preferido para trabajar usando el IDE.
  - **Modo sin proyecto**: El diseñador controla todo el proceso mediante comandos TCL.
    - Puede ejecutarlos individualmente en el TCL Shell de la herramienta.
    - Puede incluirlos en un script que Vivado ejecuta en modo batch.
    - No requiere arrancar el IDE, aunque el IDE dispone de una consola para introducirlos.

# Controlando Vivado

## comandos (i)



| Project Mode                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                      | Non-Project Mode                                                                                                                                                                                                                                                                        |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| GUI                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Tcl Script                                                                                                                                                                                                                                                                                           | Tcl Script                                                                                                                                                                                                                                                                              |
| <p>Flow Navigator</p> <ul style="list-style-type: none"> <li>PROJECT MANAGER <ul style="list-style-type: none"> <li>Settings</li> <li>Add Sources</li> <li>Language Templates</li> <li>IP Catalog</li> </ul> </li> <li>IP INTEGRATOR</li> <li>SIMULATION</li> <li>RTL ANALYSIS</li> <li>SYNTHESIS <ul style="list-style-type: none"> <li>Run Synthesis</li> <li>Open Synthesized Design</li> </ul> </li> <li><b>IMPLEMENTATION</b> <ul style="list-style-type: none"> <li>Run Implementation</li> <li>Open Implemented Design</li> </ul> </li> <li>PROGRAM AND DEBUG <ul style="list-style-type: none"> <li>Generate Bitstream</li> <li>Open Hardware Manager</li> </ul> </li> </ul> | <pre> create_project ... add_files ... import_files ... ...  launch_run synth_1 wait_on_run synth_1 open_run synth_1 report_timing_summary  launch_run impl_1 wait_on_run impl_1 open_run impl_1 report_timing_summary  launch_run impl_1 -to_step_write_bitstream wait_on_run impl_1         </pre> | <pre> read_verilog ... read_vhdl ... read_ip ... read_xdc ... read_edif ... ...  synth_design ... report_timing_summary write_checkpoint  opt_design write_checkpoint place_design write_checkpoint route_design report_timing_summary write_checkpoint  write_bitstream         </pre> |

X12974-070621

# Controlando Vivado

## ligaduras (i)



- Las ligaduras condicionan el resultado de las distintas fases del diseño:
  - Se usan **comandos TCL** para especificarlas.
  - En el **modo con proyecto** se incluyen en uno o varios **ficheros XDC**.
  - En el **modo sin proyecto** se incluyen en los **scripts** como un comando más.
- **Ligaduras de síntesis:** controlan el proceso VHDL → netlist
  - **Atributos RTL:** permiten elegir la proyección preferida de distintos elementos del diseño, la preservación de los mismos y el control de la jerarquía.
  - **Ligaduras de temporización:** permiten indicar el rendimiento esperado del circuito así como definir las características temporales del entorno en donde se integrará.
- **Ligaduras de implementación:** controlan el proceso netlist → FPGA
  - Permiten indicar la ubicación y características de los distintos elementos de diseño.
- **Ligaduras de configuración:** controlan el proceso FPGA → bitstream
  - Permiten definir las características del bitstream y del mecanismo de volcado.

# Controlando Vivado

## ligaduras (ii)



- Toda ligadura debe referirse a algún **objeto de diseño**, que puede ser:
  - **Cell**: instancia en la netlist de una primitiva de la FPGA (FFD, BRAM, DSP...)
  - **Pin**: punto de conectividad de una celda
  - **Port**: punto de conectividad de un diseño
  - **Net**: cable de interconexión entre pines o ports
  - **Clock**: señal periódica que se propaga entre los elementos síncronos.
- Los objetos heredan sus nombres de los nombres VHDL de las señales que especifican registros y puertos:
  - En registros de 1 bit: ***nombreVHDL\_reg***
  - En registros multibit: ***nombreVHDL\_reg[i]***
  - En diseños jerárquicos: ***inst\_A/inst\_B/control\_reg***
- Para poder obtener el objeto de diseño a partir de su nombre se usan los comandos `get` (que pueden usar caracteres comodín `?` y `*`):

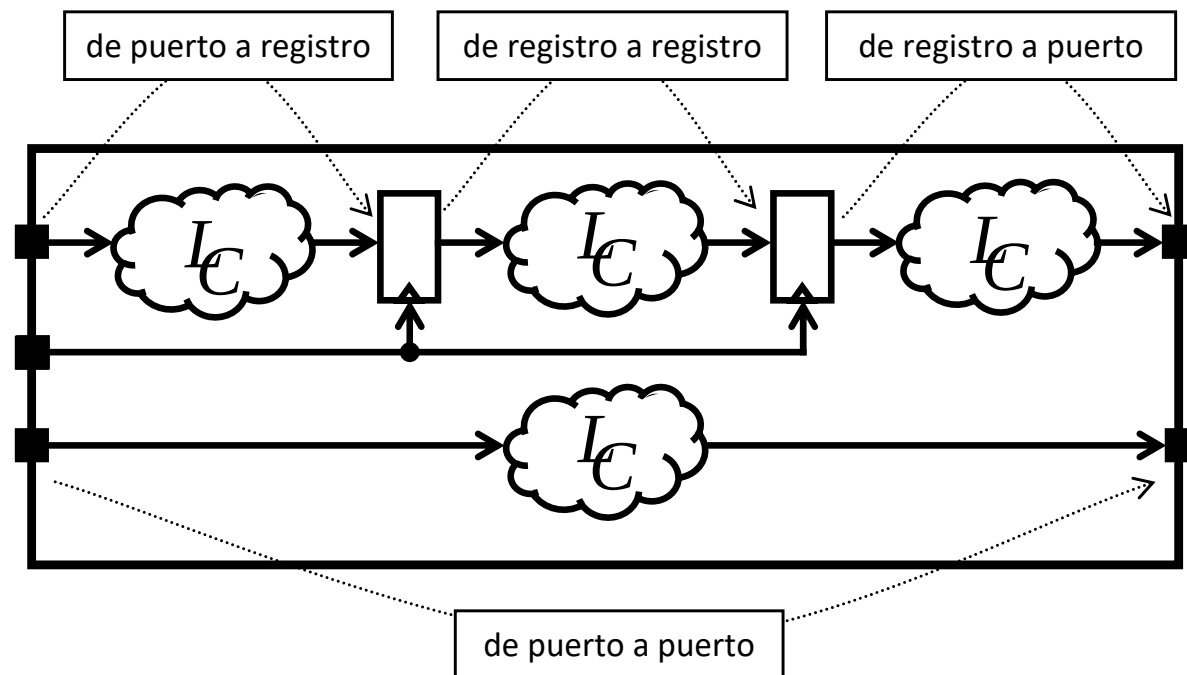
```
get_tipoObjeto patrón
```

# Controlando Vivado

## ligaduras de temporización (i)



- En un circuito de nivel lógico pueden existir 4 tipos de caminos:



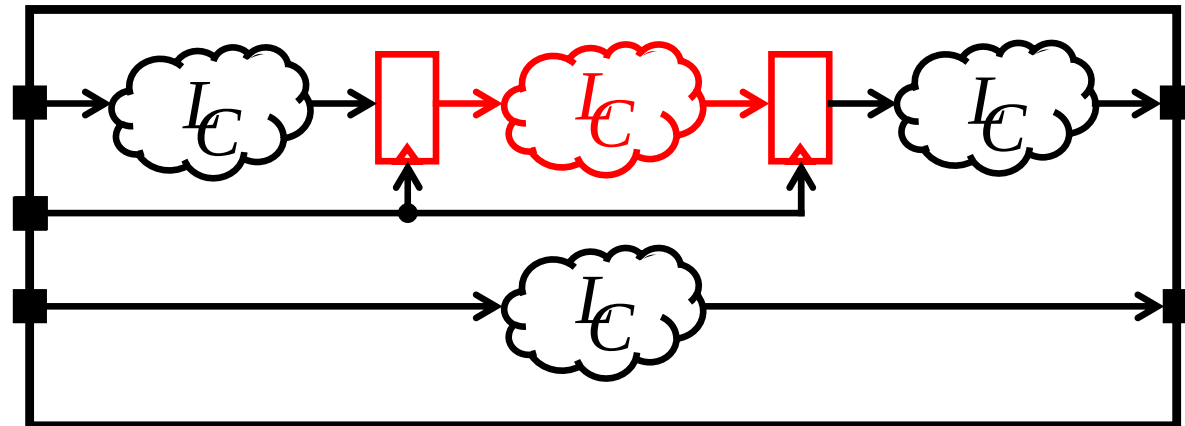
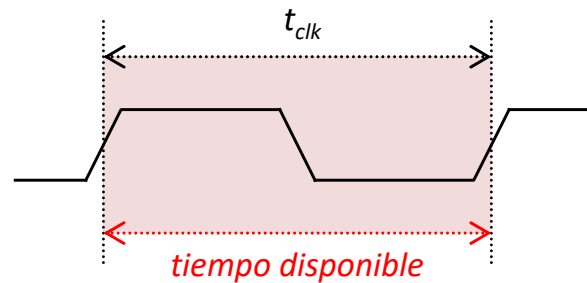
- Las ligaduras de temporización limitan el retardo max/min que podrá tener la lógica sintetizada por la herramienta en cada camino.
  - Se denomina *slack* (margen) a la diferencia entre el retardo requerido por un camino y el retardo de la lógica que atraviesa.
  - Un *slack* positivo indica que la ligadura se cumple, un *slack* negativo que no.

# Controlando Vivado

## ligaduras de temporización (ii)



- El retardo de cada tipo de camino se limita de una manera diferente:
  - Registro a registro: indicando el periodo del reloj.

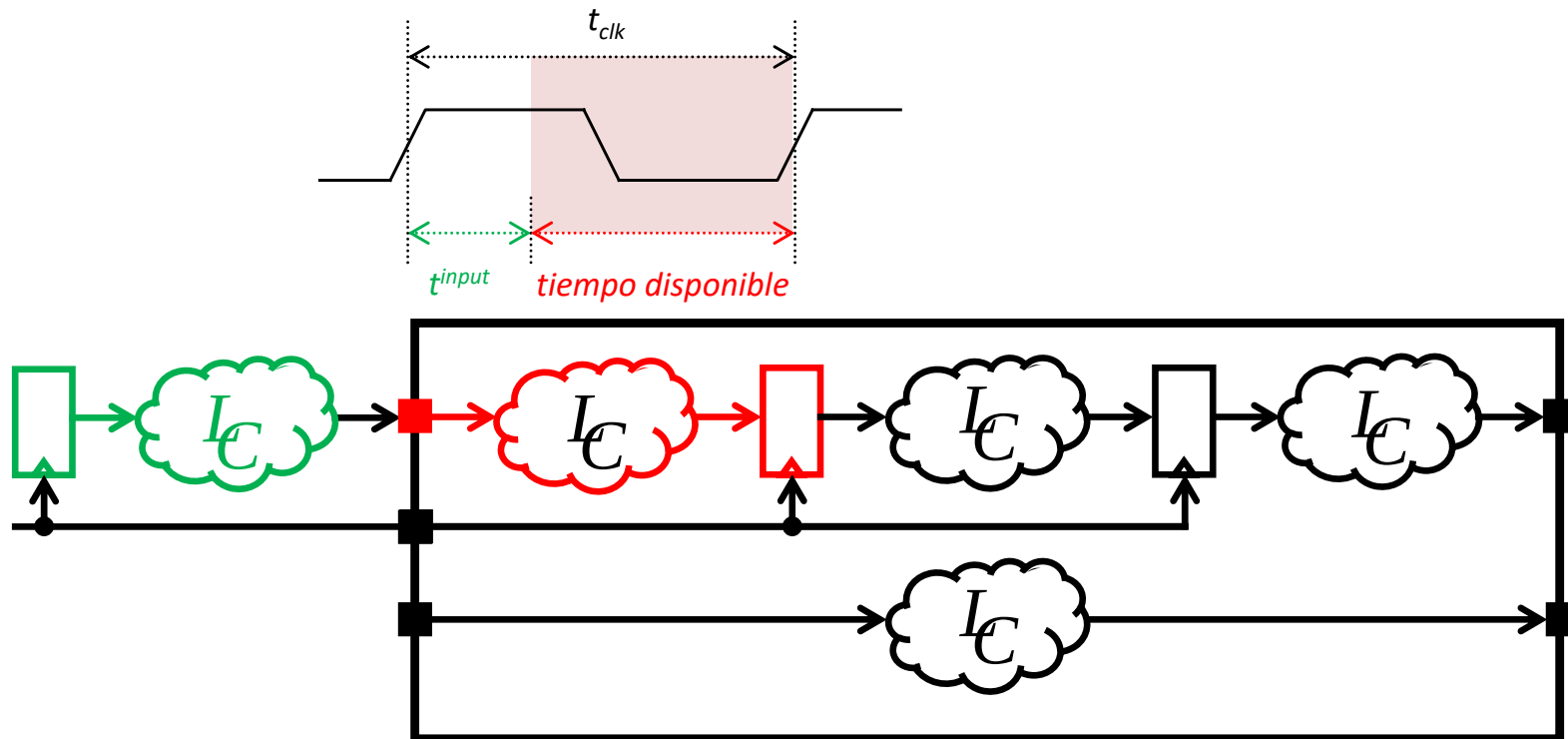


# Controlando Vivado

## ligaduras de temporización (iii)



- El retardo de cada tipo de camino se limita de una manera diferente:
  - Registro a registro: indicando el periodo de reloj.
  - Puerto a registro: indicando también el retardo de la lógica que estimula al puerto.

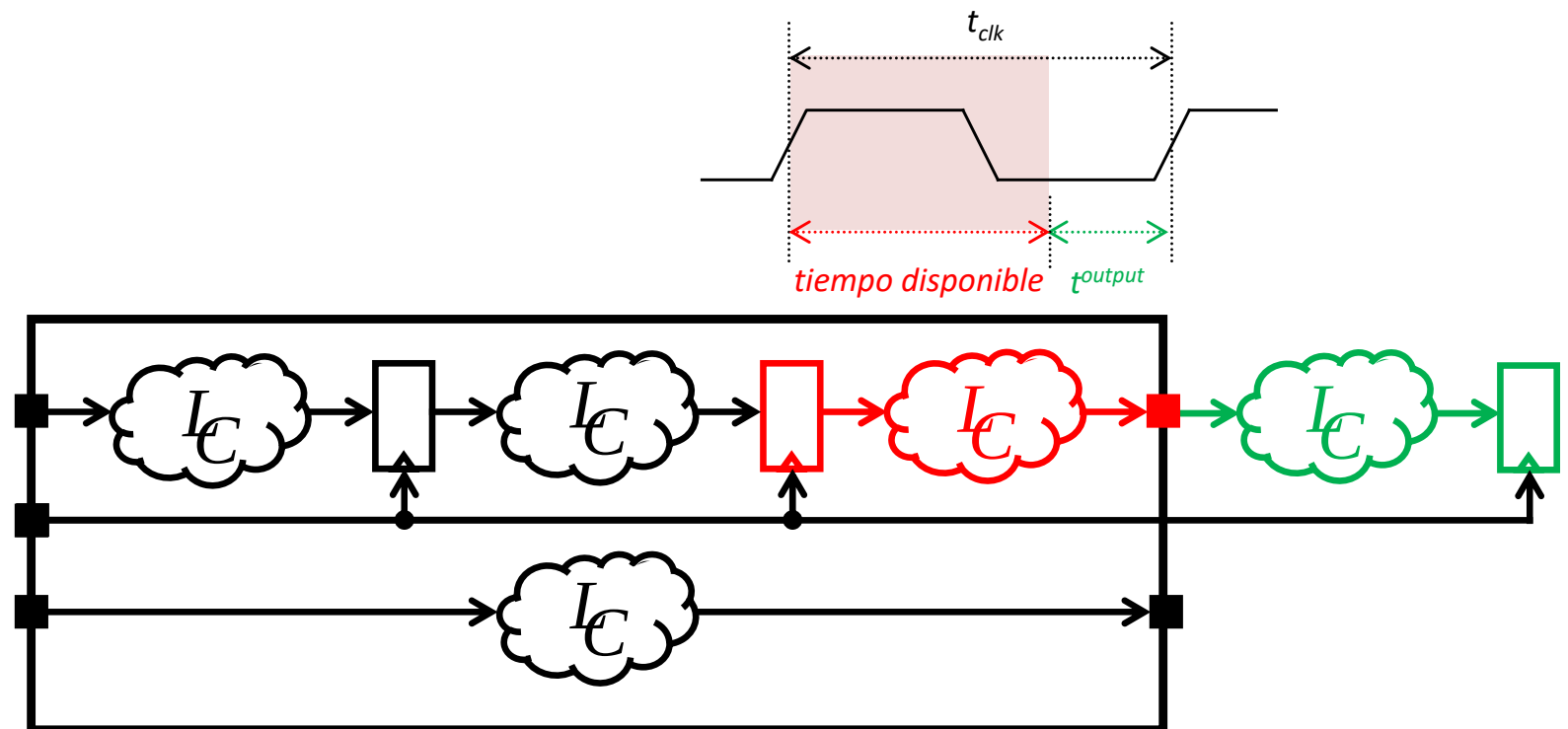


# Controlando Vivado

## ligaduras de temporización (iv)



- El retardo de cada tipo de camino se limita de una manera diferente:
  - Registro a registro: indicando el periodo de reloj.
  - Puerto a registro: indicando también el retardo de la lógica que estimula al puerto.
  - Registro a puerto: indicando también el retardo de la lógica estimulada por el puerto.

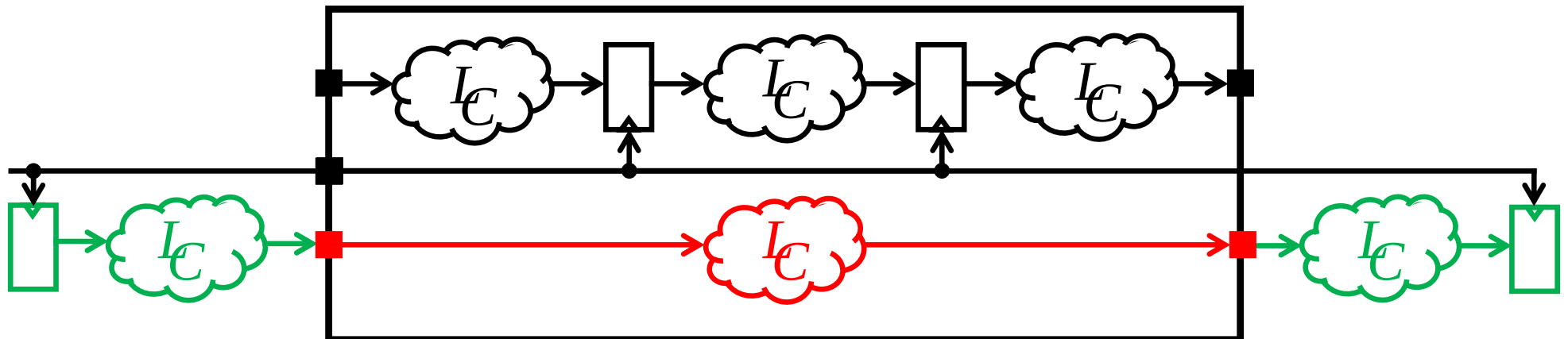
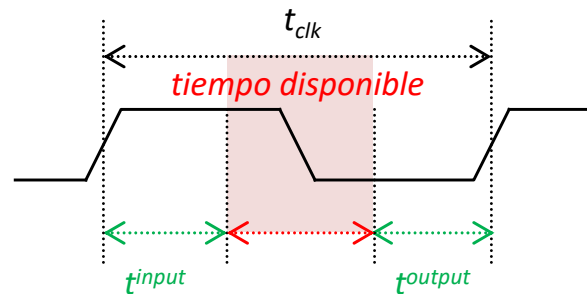


# Controlando Vivado

## ligaduras de temporización (v)



- El retardo de cada tipo de camino se limita de una manera diferente:
  - Registro a registro: indicando el periodo de reloj.
  - Puerto a registro: indicando también el retardo de la lógica que estimula al puerto.
  - Registro a puerto: indicando también el retardo de la lógica estimulada por el puerto.
  - Puerto a puerto: indicando un retardo absoluto o usando un reloj virtual.



# Controlando Vivado

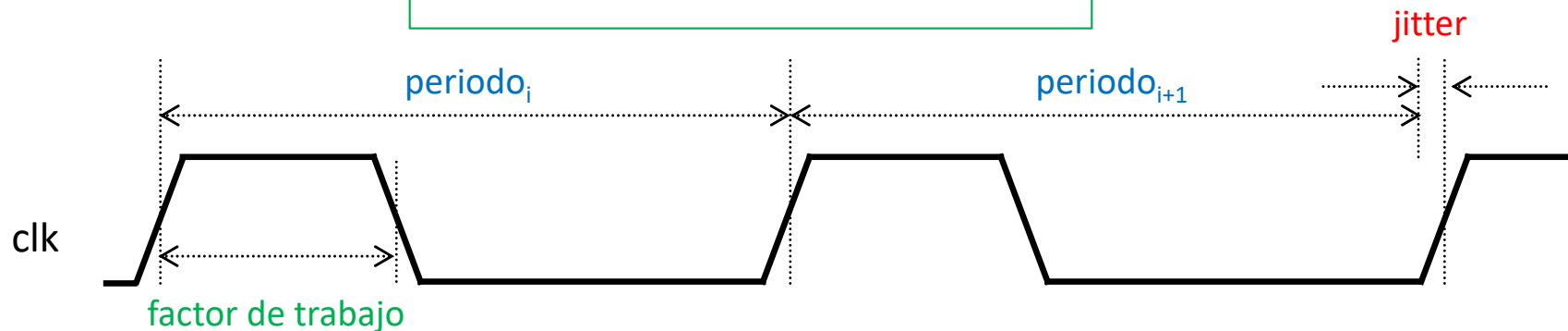
## ligaduras de temporización (vi)



- Especificación de las características de la **señal de reloj maestro**:
  - **Factor de trabajo**: Fracción del tiempo de ciclo en el que la señal de reloj está en alta.
  - **Jitter**: Desviación aleatoria máxima entre la duración de un ciclo de reloj cualquiera y la duración media del ciclo de reloj.

```
create_clock -name id -period periodo -waveform {num num} [puerto];
```

```
set_input_jitter id tiempo;
```



- Especificación de las características de un **reloj derivado**:
  - **Reloj derivado**: el que se obtiene dividiendo y/o multiplicando un reloj maestro.

```
create_generated_clock -name id -source id -divide_by num -multiply_by num;
```

# Controlando Vivado

## ligaduras de temporización (vii)



- Especificación de las características de un **reloj virtual**:

```
create_clock -name id -period periodo -waveform {num num};
```

- Especificación de **retardo externo** de una **entrada**:

```
set_input_delay -clock id -max tinput [puerto];  
set_input_delay -clock id -min tinput [puerto];
```

- Especificación de **retardo externo** de una **salida**:

```
set_output_delay -clock id -max toutput [puerto];  
set_output_delay -clock id -min toutput [puerto];
```

- Especificación de **retardo absoluto** entre 2 puertos:

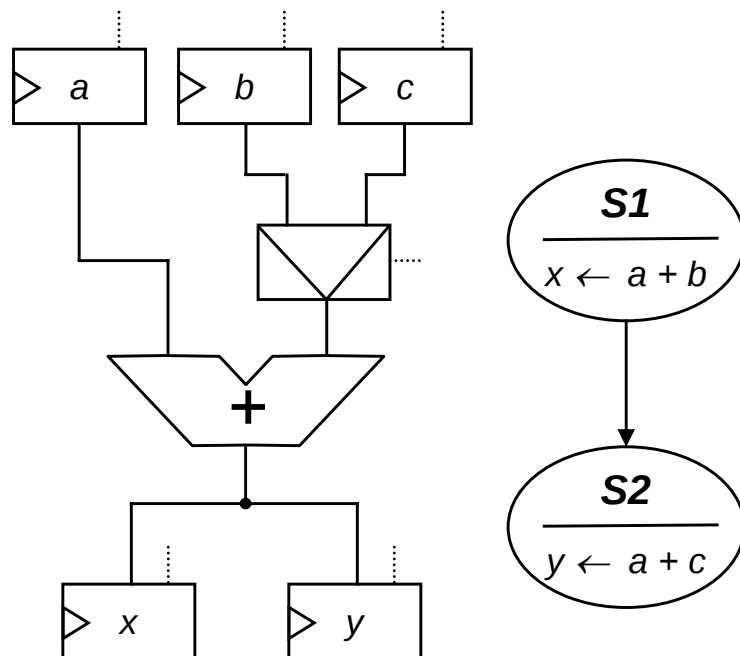
```
set_max_delay delay -from [puerto] -to [puerto];  
set_min_delay delay -from [puerto] -to [puerto];
```

# Controlando Vivado

## ligaduras de temporización: excepciones (i)



- Se denomina **camino falso** a todo camino de un circuito que sea utilizado para propagar señales inútiles.
  - No tiene que ser optimizado bajo los criterios impuestos al resto del diseño.
  - Cualquier diseño con porciones de lógica compartida puede tener caminos falsos.
  - También son caminos falsos los lazos existentes en puertos bidireccionales, los caminos que atraviesan lógica de reset, etc.



- Desde el punto de vista lógico en esta ruta de datos existen 6 caminos registro a registro:
  - $a \rightarrow x, a \rightarrow y, b \rightarrow x, b \rightarrow y, c \rightarrow x, c \rightarrow y$
- Sin embargo, si sobre ella sólo se realizan las transferencias entre registros indicadas por la máquina de estado, sólo algunos caminos son utilizados para realizar transferencias útiles:
  - $a \rightarrow x, a \rightarrow y, b \rightarrow x, c \rightarrow y$
- Luego los restantes caminos, son caminos falsos que no requieren ser optimizados:
  - $b \rightarrow y, c \rightarrow x$



# Controlando Vivado

## ligaduras de temporización: excepciones (ii)

- Se denomina **camino multiciclo** a todo camino combinacional que tarda en propagar valores más de un ciclo.
  - Es una técnica usada para solventar el problema de los grupos de lógica de retardo elevado sin reducir la frecuencia de reloj.
  - Requiere que se diseñe para que el registro ubicado al final del camino no cargue nuevos valores en todos los ciclos.

- Identificación de **caminos falsos**:

```
set_false_path -from objeto -to objeto
```

- Identificación de **caminos multiciclo**:

```
set_multicycle_path num -setup -from objeto -to objeto;  
set_multicycle_path num-1 -hold -from objeto -to objeto;
```

# Controlando Vivado

## ligaduras de implementación



- Especificación de **condiciones de funcionamiento**:
  - Las variaciones en la temperatura ambiente o voltaje de alimentación pueden alterar el retardo de un circuito.
  - Las herramientas usan en los cálculos de cualquier retardo factores de escala dependientes de las condiciones de funcionamiento previstas.

```
set_operating_conditions -voltage num -ambient_temp num;
```

- Especificación de **localización física** de un puerto:

```
set_property PACKAGE_PIN pin [puerto];
```

- Especificación de **interfaz eléctrico** de un puerto:

- Donde el interfaz depende del dispositivo: LVTTL, LVCMOS33, LVCMOS25...

```
set_property IOSTANDARD interfaz [puerto];
```

# Controlando Vivado

## ligaduras de configuración



- Especificación del interfaz de configuración:
  - Donde los voltajes del banco de configuración pueden ser: **VCC0** o **GND**.
  - Donde los voltajes de configuración pueden ser: **1.8, 2.5, 3.3...**
  - Donde los modos pueden ser: **SPIx4, SPIx2, S\_SERIAL...**

```
set_property CFGBVS valor [current_design];  
set_property CONFIG_VOLTAGE num [current_design];  
set_property CONFIG_MODE valor [current_design];
```

# Controlando Vivado

## ejemplos (i)



- Indicar que por el puerto **clk** entra una señal de reloj a 100 MHz con factor de trabajo del 50%:

```
create_clock -name sysClk -period 10.0 -waveform {0 5} [get_ports clk];
```

- Indicar que la señal que entra por el puerto **x** cambia en todos los ciclos con un retraso máximo de 3 ns respecto al flanco de subida anterior reloj:

```
set_input_delay -clock sysClk -max 3 [get_ports x];
```

- Indicar que la señal que sale por el puerto **z** requiere un margen máximo de 5 ns respecto al flanco de subida del anterior reloj:

```
set_input_delay -clock sysClk -max 2 [get_ports z];
```

- Indicar que la señal **clk1** (12.5 MHz) se obtiene por división de la frecuencia del reloj anterior:

```
create_generated_clock -name clk1 -source sysClk -divide_by 8;
```

# Controlando Vivado

## ejemplos (ii)



- Indicar que el retardo combinacional máximo entre los puertos **x** y **z** es 5 ns:

```
set_max_delay 5 -from [get_ports x] -to [get_ports z];
```

- Indicar lo anterior usando un reloj virtual:

```
create_clock -name vClk -period 10.0  
set_input_delay -clock vClk -max 2.5 [get_ports x];  
set_output_delay -clock vClk -max 2.5 [get_ports z];
```

- Indicar que todos los caminos que unen el **regA** y el **regB** son falsos:

```
set_false_path -from [get_pins regA*/C] -to [get_pins regB*/D]
```

- Indicar que **regB** solo carga valores de **regA** en 1 de cada 2 ciclos, es decir, que el camino que une **regA** y **regB** es multiciclo (de 2 ciclos):

```
set_multicycle_path 2 -setup -from [get_pins regA*/C] -to [get_pins regB*/D]  
set_multicycle_path 1 -hold -from [get_pins regA*/C] -to [get_pins regB*/D]
```

# Controlando Vivado

## ejemplos (iii)



- Indicar condiciones ambientales:

```
set_operating_conditions -ambient_temp 10
```

- Indicar el interfaz eléctrico del puerto x:

```
set_property IOSTANDARD LVCMOS33 [get_ports x];
```

- Indicar que el puerto **clk** se ubique en el pin W5 de la FPGA:

```
set_property PACKAGE_PIN W5 [get_ports clk];
```

- Indicar el interfaz de configuración:

```
set_property CFGBVS VCC0 [current_design];  
set_property CONFIG_VOLTAGE 3.3 [current_design];  
set_property CONFIG_MODE SPIx4 [current_design];
```



# Acerca de *Creative Commons*



## ■ Licencia CC (*Creative Commons*)

o Ofrece algunos derechos a terceras personas bajo ciertas condiciones. Este documento tiene establecidas las siguientes:



**Reconocimiento** (*Attribution*):

En cualquier explotación de la obra autorizada por la licencia hará falta reconocer la autoría.



**No comercial** (*Non commercial*):

La explotación de la obra queda limitada a usos no comerciales.



**Compartir igual** (*Share alike*):

La explotación autorizada incluye la creación de obras derivadas siempre que mantengan la misma licencia al ser divulgadas.

Más información: <https://creativecommons.org/licenses/by-nc-sa/4.0/>