

Outils de simulation VHDL

INF3500

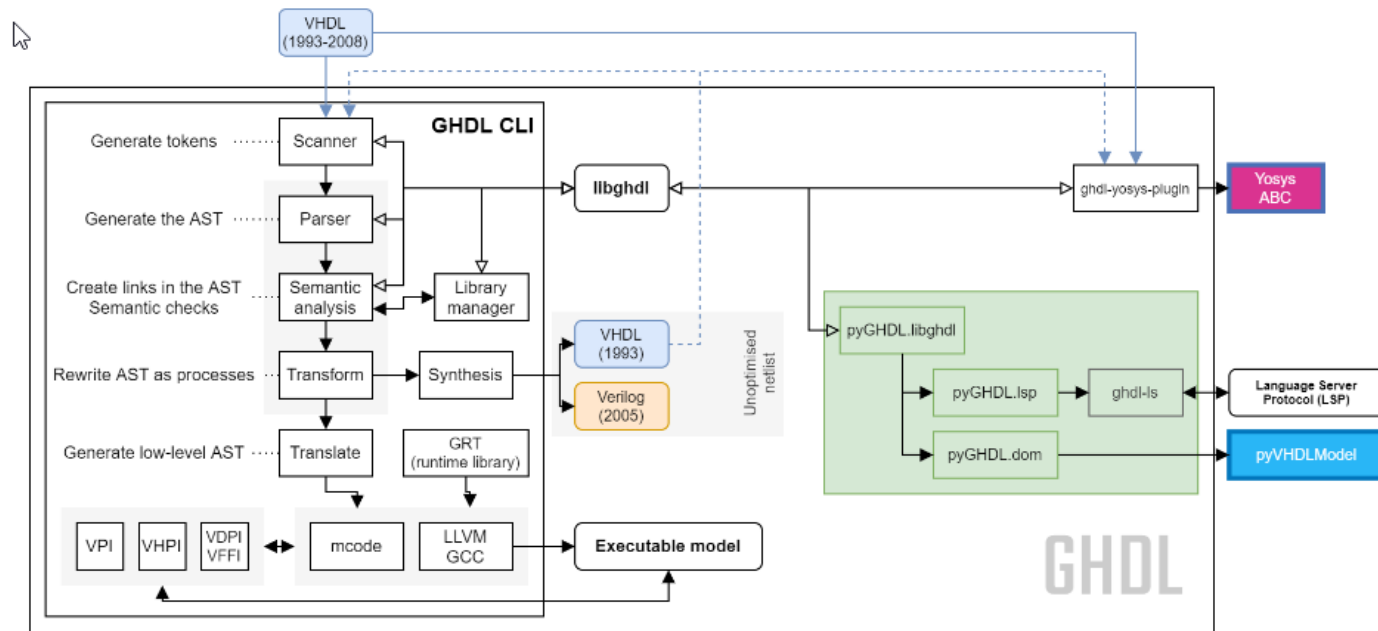
Conception et réalisation de systèmes numériques



**POLYTECHNIQUE
MONTREAL**

GHDL et gtkWave (simulation VHDL)

- Vous pouvez installer ghdl et gtkwave sur Windows avec des versions Windows. Vous pouvez aussi vous installer une distribution linux sur votre PC Windows et pouvoir rouler GHDL et mode CLI sur Linux.



GHDL : Windows

Installation de GHDL et gtkWave

GHDL

- Aller sur le site suivant : <https://github.com/ghdl/ghdl> et trouver la version la plus récente pour Windows (par exemple : ghdl-MINGW32.zip).
- Créer un répertoire EDA sur le C de votre ordinateur
- Ouvrir le fichier ZIP et extraire le répertoire GHDL pour le mettre dans votre répertoire EDA.
- Pour partir l'outil, vous devez aller dans le répertoire bin/ et utiliser l'exécutable : [gtkwave.exe](#)

gtkWave

- Pour visualiser vos signaux de simulation (wave) vous devrez utiliser gtkWave (<https://sourceforge.net/projects/gtkwave/> mais attention de bien aller chercher la version Windows [gtkwave-3.3.98-bin-win64.zip](#)).
- Ouvrir le fichier ZIP et extraire le répertoire gtkWave64 pour le mettre dans votre répertoire EDA.
- Pour partir l'outil, vous devez aller dans le répertoire bin/ et utiliser l'exécutable : [gtkwave.exe](#) (vous pouvez vous créer un raccourci)

GHDL : Windows

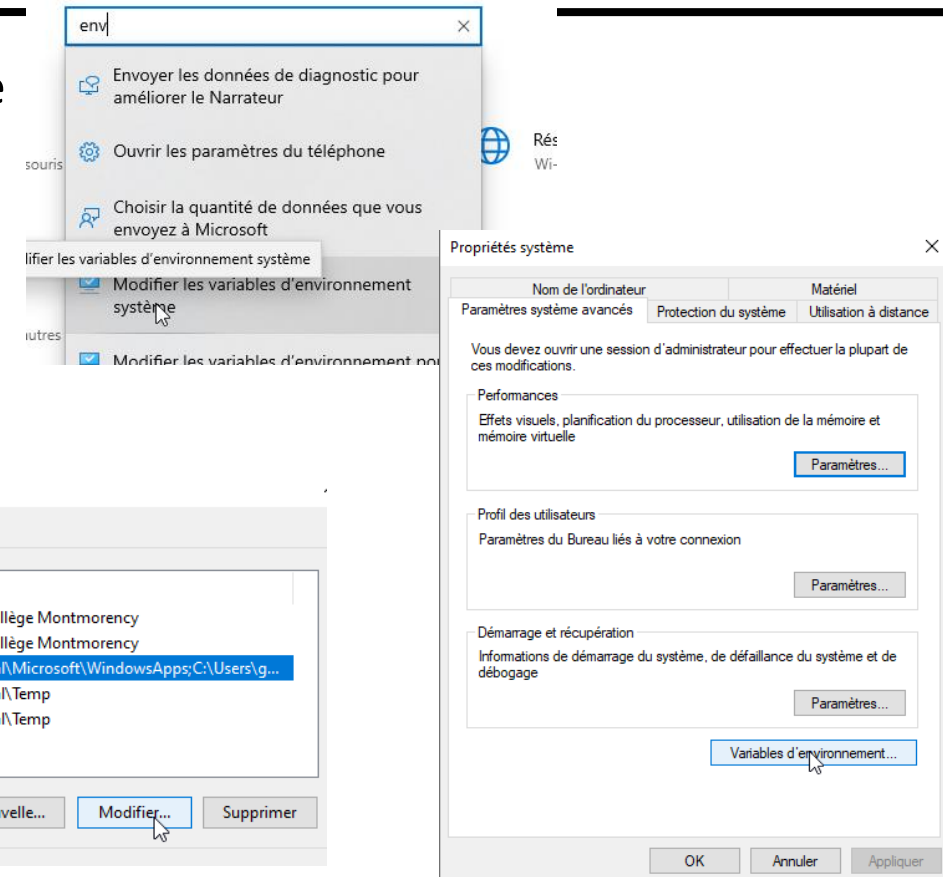
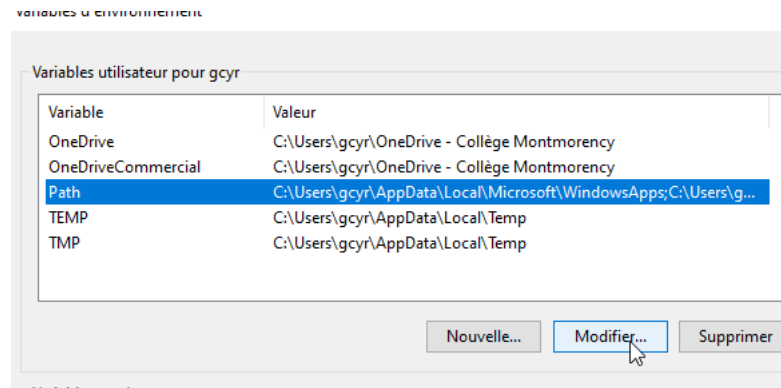
Modifier la variable PATH

Pour faciliter l'utilisation des exécutables sur Windows (interface ligne de commande), il est préférable d'ajouter le chemin des exécutables dans la variable PATH de Windows.

- Dans le « settings » de l'ordinateur, chercher et choisir : « Modifier les variables d'environnement système »
- Puis choisir « variable d'environnement »
- Modifier la variable « Path »
- Ajouter le chemin pour GHDL

Modifier la variable d'environnement

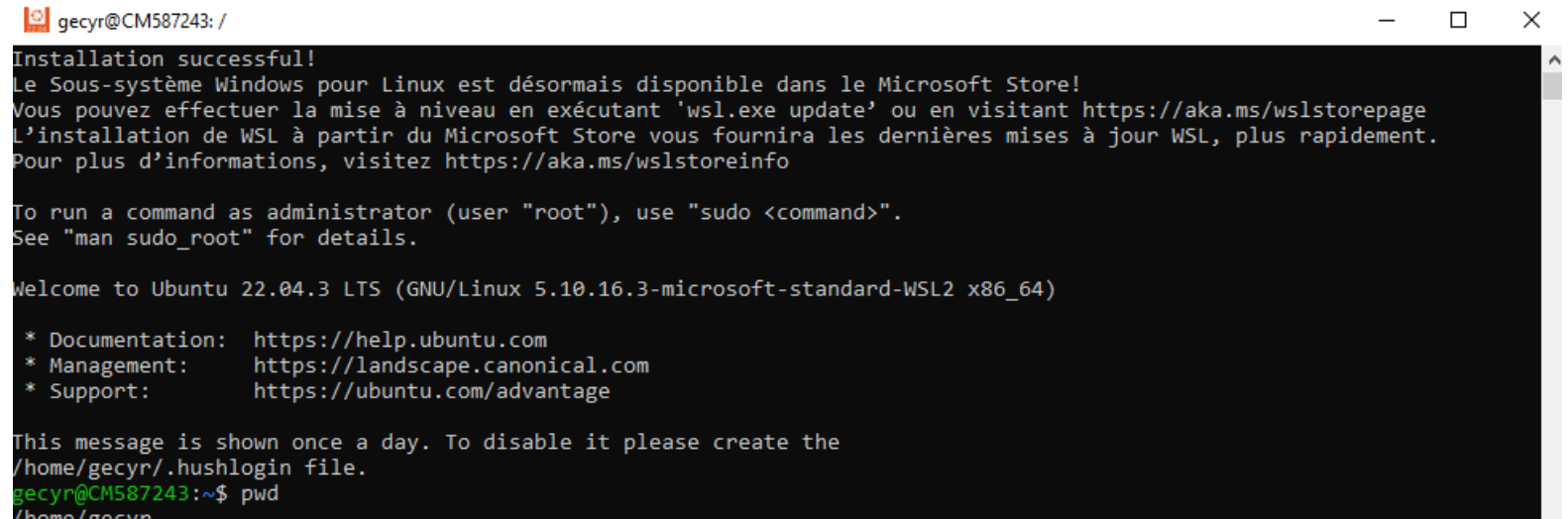
```
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps  
C:\Users\gcyr\AppData\Local\Programs\Microsoft VS Code\bin  
C:\EDA\GHDL\bin
```



GHDL : Linux

Installation d'une distribution Linux sur Windows

- Pour installer une distribution Linux sur Windows, vous pouvez utiliser WSL (Windows Subsystem for Linux) sur Windows. Pour l'installer, vous pouvez aller dans PowerShell en administrateur et taper : `wsl.exe – install`.
- Si cela ne fonctionne pas, suivez les étapes données à la page suivante pour le faire manuellement.



```
gecyr@CM587243: /
Installation successful!
Le Sous-système Windows pour Linux est désormais disponible dans le Microsoft Store!
Vous pouvez effectuer la mise à niveau en exécutant 'wsl.exe update' ou en visitant https://aka.ms/wslstorepage
L'installation de WSL à partir du Microsoft Store vous fournira les dernières mises à jour WSL, plus rapidement.
Pour plus d'informations, visitez https://aka.ms/wslstoreinfo

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.10.16.3-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

This message is shown once a day. To disable it please create the
/home/gecyr/.hushlogin file.
gecyr@CM587243:~$ pwd
/home/gecyr
```

GHDL : Linux

Installer le WSL sur Windows manuellement

- Ouvrir PowerShell en mode administrateur puis faire la commande suivante et redémarrer l'ordinateur.

```
Administrateur : Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\WINDOWS\system32> Enable-WindowsOptionalFeature -Online -FeatureName VirtualMachinePlatform
Voulez-vous redémarrer l'ordinateur pour terminer cette opération maintenant ?
[Y] Yes [N] No [?] Aide (la valeur par défaut est « Y ») : █
```

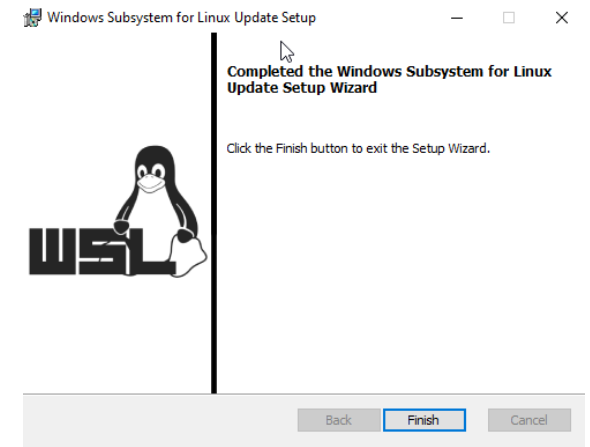
- Ouvrir PowerShell en mode administrateur puis faire la commande suivante et redémarrer l'ordinateur.

```
PS C:\WINDOWS\system32> Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux
Voulez-vous redémarrer l'ordinateur pour terminer cette opération maintenant ?
[Y] Yes [N] No [?] Aide (la valeur par défaut est « Y ») : █
```

- Télécharger le package de mise à jour du noyau Linux : wsl_update_x64.msi

- Installer wls :

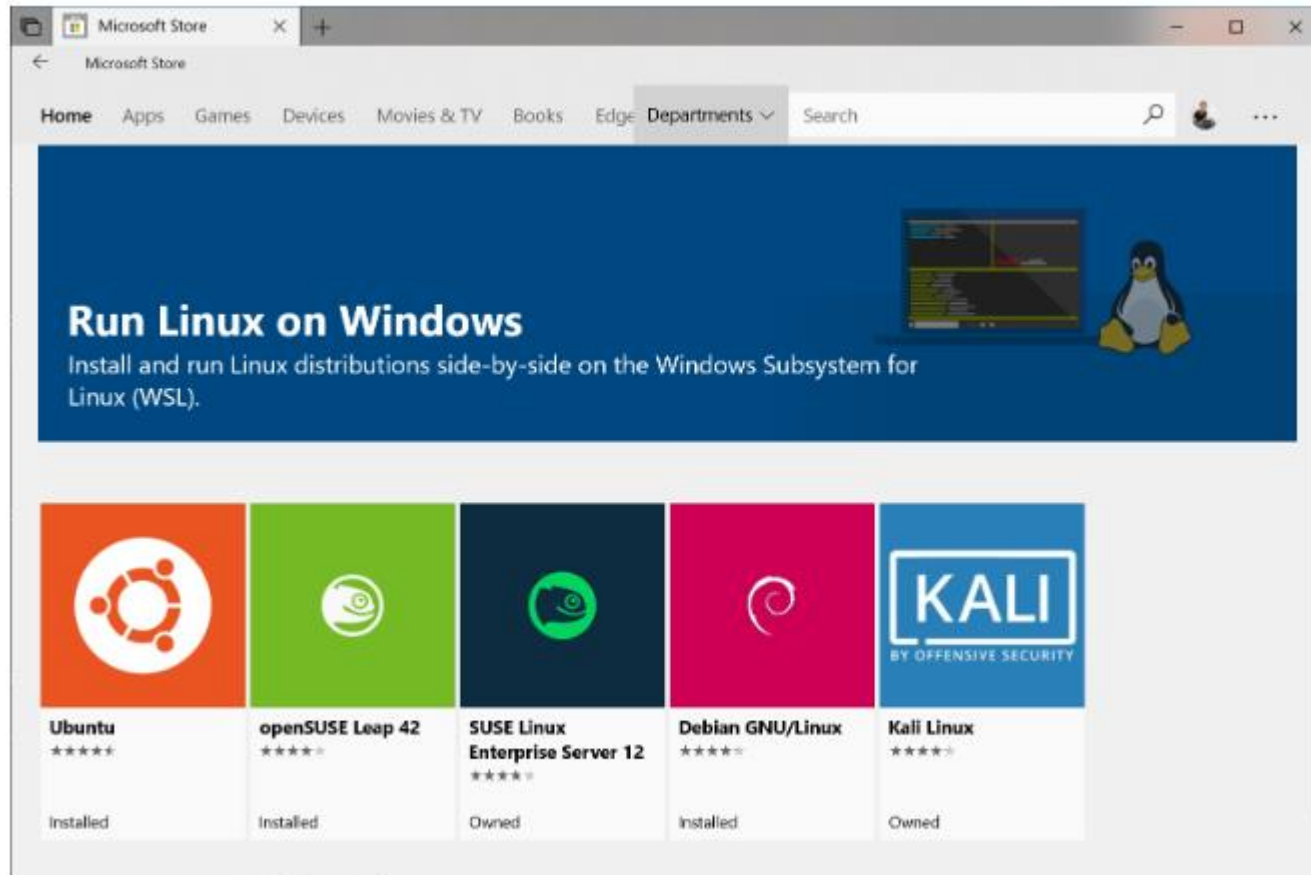
```
PS C:\Users\gcyr> wsl.exe --set-default-version 2
Pour plus d'informations sur les différences de clés avec WSL 2, visitez https://aka.ms/wsl2
L'opération a réussi.
PS C:\Users\gcyr> █
```



GHDL : Linux

Installer une distribution Linux pour Windows

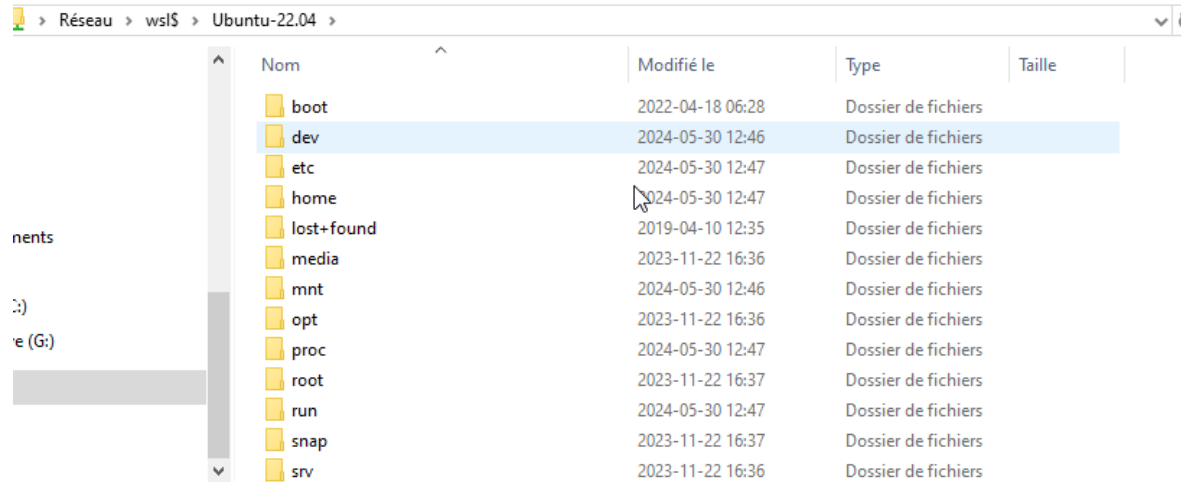
- Choisir votre distribution Linux et l'installer (vous pouvez choisir Ubuntu 20.04)



GHDL : Linux

Accéder aux fichiers de Linux à Windows et Windows à Linux

- Pour accéder aux fichiers Linux de l'explorateur Windows taper : `\\wsl$` dans la barre de navigation



- Pour accéder aux répertoire Windows sous Linux taper : `cd /mnt/c`

```
gecyr@CM587243:/mnt/c$ ls
ls: cannot access 'DumpStack.log.tmp': Permission denied
ls: cannot access 'hiberfil.sys': Permission denied
ls: cannot access 'pagefile.sys': Permission denied
ls: cannot access 'swapfile.sys': Permission denied
$Recycle.Bin          config.Msi           KRAMER               ProgramData           Xilinx
$WINDOWS.BOOT        del                  My_Designs           Python27              hiberfil.sys
$WinREAgent          'Documents and Settings'  OneDriveTemp         Recovery              inetpub
$Windows.BootCache  DumpStack.log        PerfLogs             'System Volume Information'  maxima-5.47.0
Alder                DumpStack.log.tmp   PolyGcylLocal        FTP-Root              msdia80.dll
Brother              ESD                 Program Files         Users                  pagefile.sys
CommunicatorInstall.log  Inta                Program Files (x86)  Windows                swapfile.sys
gecyr@CM587243:/mnt/c$
```


GHDL : Linux

Installer les packages sur Linux

- Si vous avez installé Ubuntu 20.04, vous aurez besoin de certains packages qui ne sont pas disponibles. Sur le Shell Linux entrer les trois commandes suivantes :

```
sudo apt install make
```

```
sudo add-apt-repository universe
```

```
sudo apt update
```

```
sudo apt install csh
```

```
sudo apt install tclsh
```

```
sudo apt install tk8.6-dev
```

```
sudo apt install tcl8.5-dev
```

```
sudo apt install -y
```

```
sudo apt install -y gpert
```

```
sudo apt install libbz2-dev
```

```
sudo apt install libgtk2.0-dev libglib2.0-dev
```

GHDL : Linux

Installation de GHDL et gtkWave

GHDL

- GHDL est nativement utilisé sur Linux, mais peut aussi être installé sur Windows, et utiliser par une interface ligne de commande.
- Vous pouvez télécharger et installer l'installateur pour Windows. La version la plus simple à installer sur WSL est la version « ghdl-installer-0.29.1.exe » sur le site : <http://ghdl.free.fr/download.html>. Ce n'est pas la plus récente (elle ne supporte pas le VHDL 2008), mais elle est facile à installer puisque c'est un exécutable Windows.
- Par la suite, vous devrez faire un Makefile pour compiler vos fichiers et l'exécuter sur votre Shell Linux.

gtkWave

- Nous allons utiliser la même version que celle qui a été installée dans la partie GHDL : Windows, donc rien de plus à faire ici.

GHDL : Linux

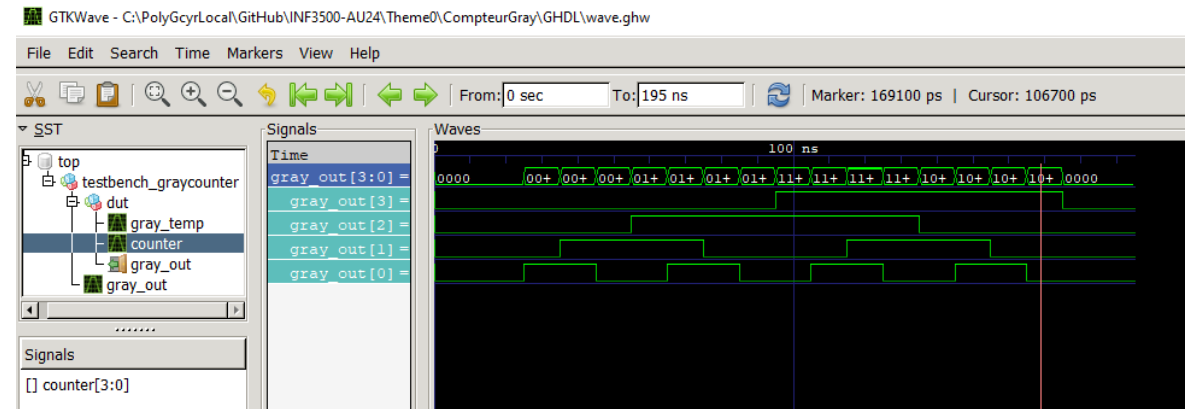
Utilisation de GHDL et gtkWave

- Par la suite, vous devrez faire un Makefile pour compiler vos fichiers et l'exécuter sur votre Shell Linux. Voici un exemple pour mon compteur gray

```
GHDL = ghdl.exe
FLAGS = "--std=93"

all:
    $(GHDL) -a $(FLAGS) GrayCounter.vhd Testbench_GrayCounter.vhd
    $(GHDL) -e $(FLAGS) Testbench_GrayCounter
    $(GHDL) -r $(FLAGS) Testbench_GrayCounter --wave=wave.ghw --stop-time=lus
```

- En allant dans le répertoire de vos fichiers, vous n'avez qu'à faire : make
- Le script fait 3 étapes :
 - Compilation (avec le -a)
 - Élaboration (avec le -e)
 - Simulation (avec le -r)
- Par la suite, un fichier .ghw sera créé et vous pourrez l'ouvrir dans gtkWave pour voir le résultat de la simulation



Active-HDL de Aldec (Simulateur HDL)

- Si vous décidez d'utiliser Active-HDL pour vos simulations, vous pouvez télécharger la version la plus récente étudiant et gratuite :
https://www.aldec.com/en/products/fpga_simulation/active_hdl_student
- Pour savoir comment créer votre projet dans Active-HDL, vous pouvez regarder les étapes dans :
<https://www.aldec.com/en/support/resources/documentation/articles/1054>



Free Active-HDL Student Edition

Active-HDL Student Edition is a mixed language design entry and simulation tool offered at no cost by Aldec for students to use during their course work.

Licensing

Active-HDL Student Edition includes a "load and go" license. This means students can begin using it immediately after installing.

Key Features of Active-HDL Student Edition

- Mixed language simulator
- Multi-FPGA & EDA Tool Design Flow Manager
- Graphical Design entry & editing
- Code2Graphics and Graphics2Code
- Pre-compiled FPGA vendor libraries
- IEEE Language Support: VHDL, Verilog, SystemVerilog(Design), SystemC
- Waveform Viewer and List Viewer
- Interface with MATLAB®/Simulink®
- HTML and PDF Design Documentation

Active-HDL

Création d'un projet

1 workspace not loaded) - Active-HDL Student Edition (64-bit)

File Edit Search View Workspace Design Simulation Tools Window Help

Open Workspace...
Close Workspace

New > Workspace...
Open... Ctrl+O Design...
Open design from Source Control
Open Symbol... VHDL Source...
Close
Close All

2 New Design Wizard

How would you like to create Design Resources?

Create an empty design
 Create an empty design with Design Flow
 Add existing resource files

This option creates an empty design, allows specifying sources to be added prior to creating the design, and enables Design Flow Manager. You can select a vendor of your synthesis or implementation tool, technology, libraries, and specify the default HDL language of your new design entry sources.

Create New Workspace
 Add Design to Current Workspace

< Précédent Suivant > Annuler

3 New Design Wizard

Select resource files to be added to the design.

The following files will be added to the design:

Path	Contents
c:\users\gcy\r\downloads\labo0\graycounter...	VHDL Source Code
c:\users\gcy\r\downloads\labo0\testbench_gr...	VHDL Source Code

< Précédent Suivant > Annuler

4 Property Page

Specify additional information about the new design.

Design Language

Block Diagram Configuration: Default HDL Language

Default HDL Language: VHDL

Target Technology

Vendor: Xilinx2023x

Technology: Zynq

< Précédent Suivant > Annuler

Active-HDL

Création d'un projet (suite)

New Design Wizard 5

Specify basic information about the new design.

Type the design name:

Select the location of the design folder:

Do not create design directory

The name of the default working library of the design:

The name specified here will be used as the file name for the logical name of the library. You can change the logical name later.

Design file path:

< Précédent **Suivant >**

New Design Wizard 6

The new design will have the following specifications:

Design name: Labo0

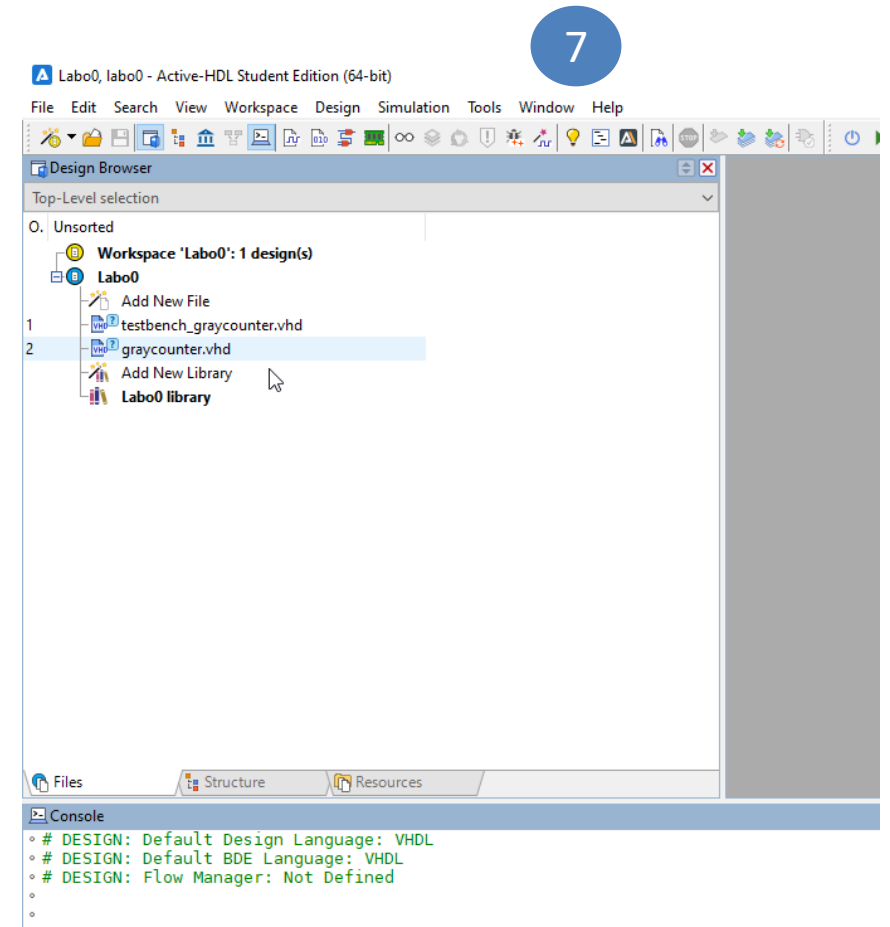
The following files will be added to the design:

- c:\users\gcyr\downloads\labo0\testbench_graycounter.vhd
- c:\users\gcyr\downloads\labo0\graycounter.vhd

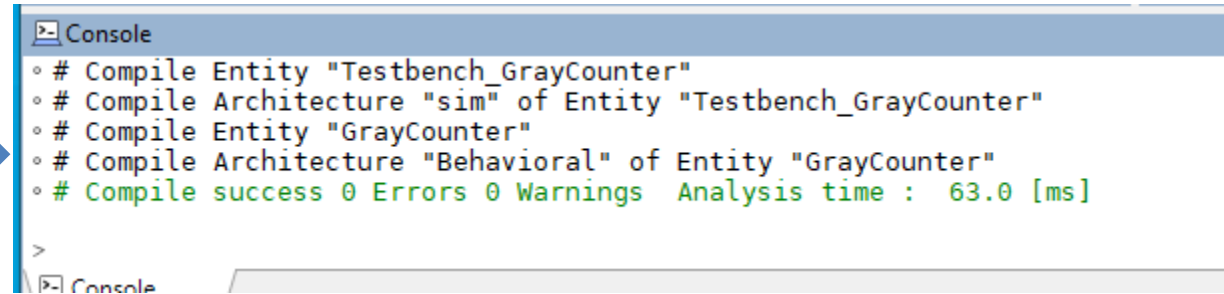
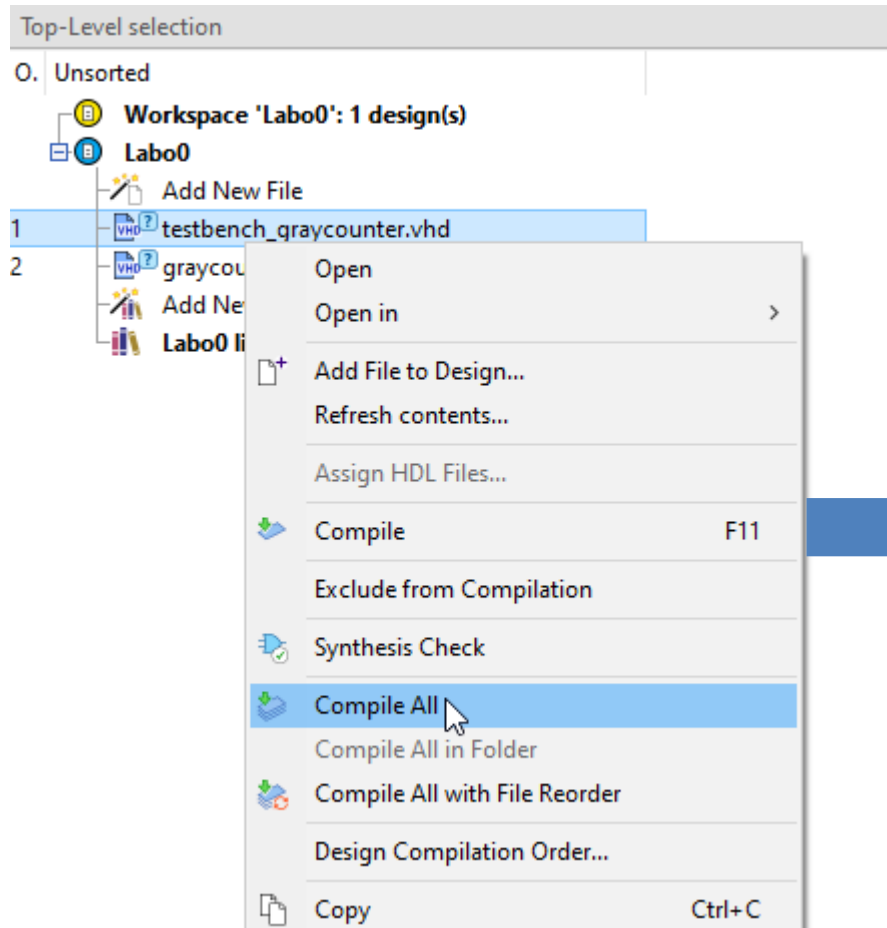
Design file path:

Compile source files after creation

< Précédent **Terminer** Annuler

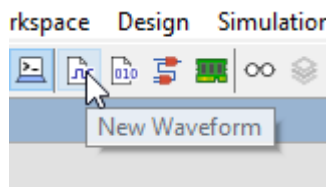


Active-HDL Compilation



Active-HDL Simulation

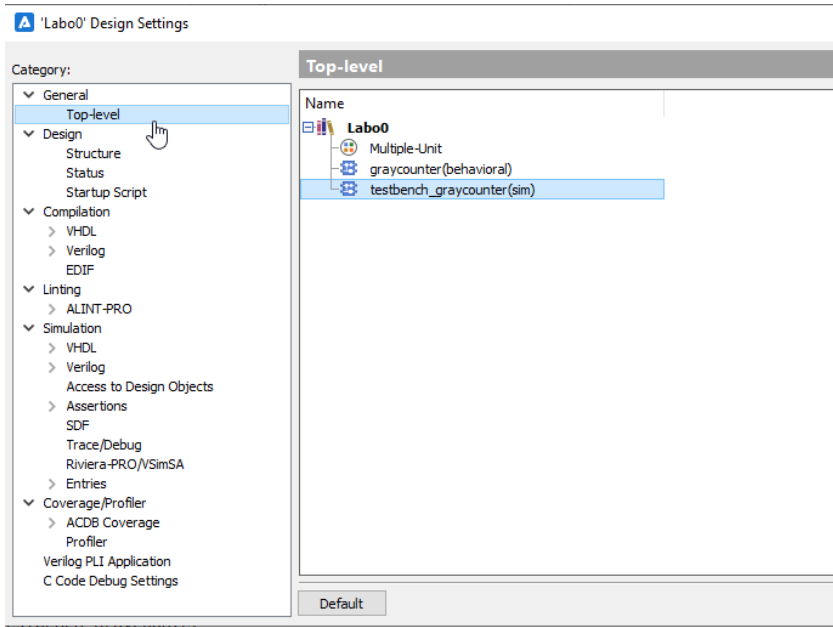
1



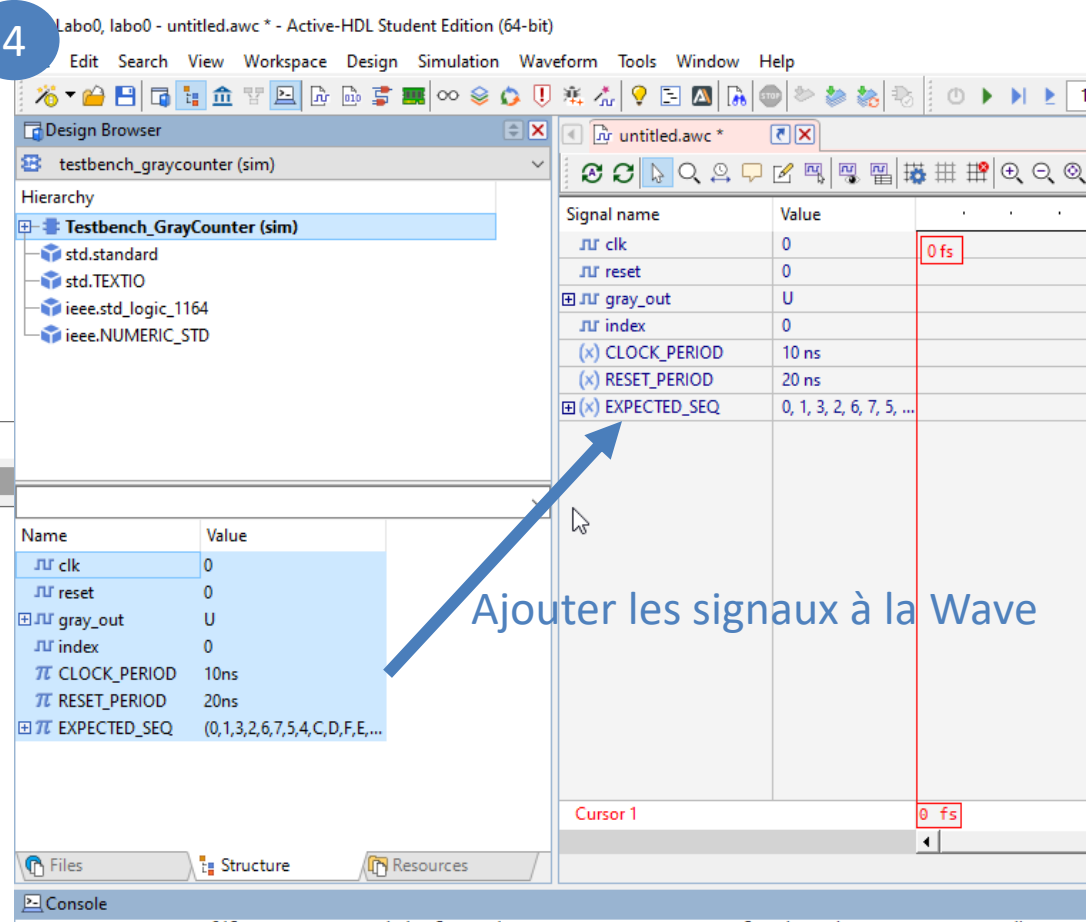
2



3

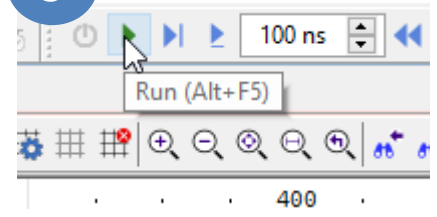


4



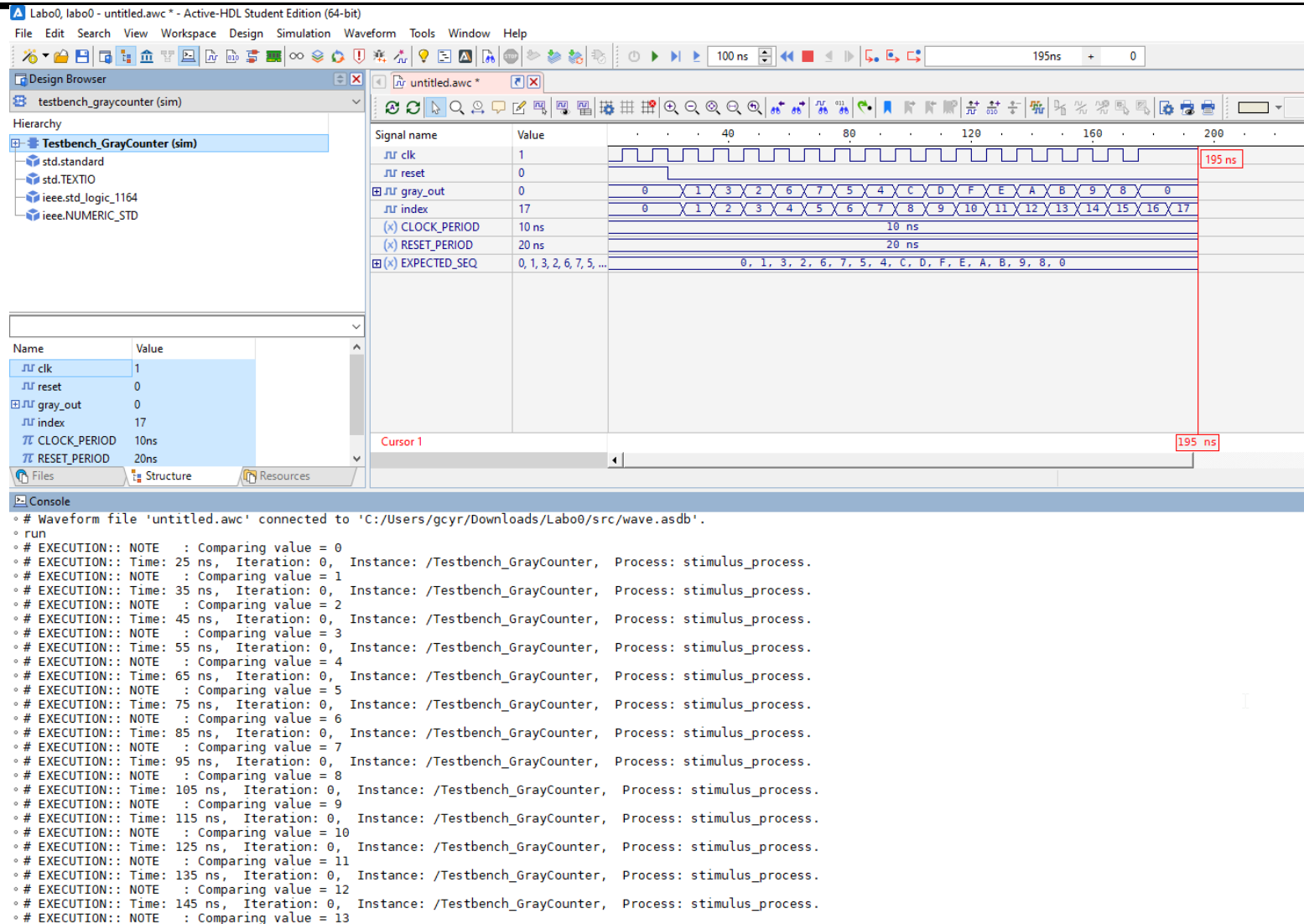
Ajouter les signaux à la Wave

5



Active-HDL

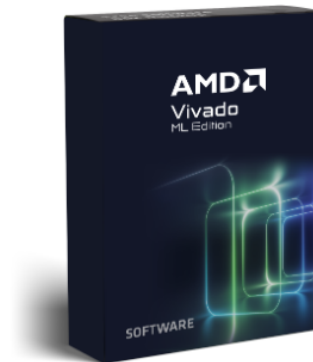
Résultats de simulation



Vivado (outil de développement Xilinx)

- Pour utiliser et installer qu'un seul outil pour la compilation, simulation, synthèse et implémentation, on vous suggère d'utiliser Vivado pour simuler.
- Vous pouvez télécharger la version la plus récente (2024.1) et gratuite
[:https://www.xilinx.com/products/design-tools/vivado/vivado-buy.html](https://www.xilinx.com/products/design-tools/vivado/vivado-buy.html)

Home / Design Tools / Vivado Overview / Vivado ML



Click to Enlarge



Vivado ML Standard & Enterprise Edition

by: AMD

Download Vivado™ ML Standard Edition free. Purchase licensing options for Enterprise Edition start at \$2995.

Download from Download Center

Buy License

Product Add-Ons:

Vitis Model Composer

Vivado Installation

1

Vous devrez d'abord vous identifier avec un compte créer sur : <https://sso-registration-qa.amd.com/> pour pouvoir vous identifier lors de l'installation par la suite.

2

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.2 - 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 © 2022-2024 Advanced Micro Devices, Inc. All rights reserved.

< Back Next > Cancel

3

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.2 - Select Product to Install

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

Vits

Installs Vits Core Development Kit for embedded software and application acceleration development on AMD platforms. Vits installation includes Vivado Design Suite. Users can also install Vits 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 Vits High-Level Synthesis, implementation, verification and device programming. Complete device support, cable driver, and Document Navigator included. Users can also install Vits Model Composer to design for AI Engines and Programmable Logic in MATLAB and Simulink. Users can select to install the Vits Embedded Development which is an embedded software development package.

Vits Embedded Development

The Vits Embedded Development is a standalone embedded software development package for creating, building, debugging, optimizing, and downloading software applications for AMD FPGA processors. It includes a new Vits IDE (Preview) with its new backend Vits Server, as well as the classic command line utilities such as hv_server, bootgen and program_flash.

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 © 2022-2024 Advanced Micro Devices, Inc. All rights reserved.

< Back Next > Cancel

4

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.2 - Select Edition to Install

Select Edition to Install

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

Vivado ML Standard

Vivado ML Standard Edition is the no-cost, device limited version of the Vivado ML Enterprise edition. Users can add Vits Model Composer which is an AMD toolbox for MATLAB and Simulink to design for AI Engines and Programmable Logic. Users can select to install the Vits Embedded Development which is an embedded software development package. If you have been using AMD System Generator for DSP, you can continue development using Vits Model Composer.

Vivado ML Enterprise

Vivado ML Enterprise Edition includes the full complement of Vivado Design Suite tools for design, including C-based design with Vits HLS, implementation, verification, and device programming. Complete device support, cable drivers, and documentation Navigator are included. Users can add Vits Model Composer which is an AMD toolbox for MATLAB and Simulink to design for AI Engines and Programmable Logic. Users can select to install the Vits Embedded Development which is an embedded software development package. If you have been using AMD System Generator for DSP, you can continue development using Vits Model Composer.

< Back Next > Cancel

Vivado Installation (suite)

1

AMD Unified Installer for FPGAs & Adaptive SoCs 2024.1 - Vivado ML Standard

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

Vivado ML Standard Edition is the no-cost, device limited version of the Vivado ML Enterprise edition. Users can add Vitis Model and Simulink to design for AI Engines and Programmable Logic. Users can select to install the Vitis Embedded Development which package. If you have been using AMD System Generator for DSP, you can continue development using Vitis Model Composer. Manager for power estimation of Versal, UltraScale+, and Kria products.

- Design Tools
 - Vivado Design Suite
 - Vivado
 - Vitis HLS
 - Vitis Model Composer (Toolbox for MATLAB and Simulink. Includes the function block editor, System Generator for DSP, Vitis Embedded Development)
 - Power Design Manager (PDM)
 - DocNav
- Devices
 - Install Devices for Kria SOMs and Starter Kits
 - Production Devices
 - SoCs
 - Zynq-7000 (limited support)
 - Zynq UltraScale+ MPSoC (limited support)
 - Zynq UltraScale+ RFSoc
 - 7 Series (limited support)
 - Artix-7
 - Kintex-7
 - Spartan-7
 - Virtex-7
 - UltraScale (limited support)
 - UltraScale+ (limited support)
 - Versal ACAP
 - Engineering Sample Devices
- Installation Options
 - Install Cable Drivers (You MUST disconnect all Xilinx Platform Cable USB II cables before installation)

Download Size: 13.2 GB
Disk Space Required: 47.81 GB

2

AMD Unified Installer for FPGAs & Adaptive SoCs 2024.1 - Accept License Agreements

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

End User License Agreement for Vivado

By checking "I Agree" below, or OTHERWISE ACCESSING, DOWNLOADING, INSTALLING or agreement, which can be viewed by [clicking here](#).

I Agree

Third Party Software End User License Agreement for Vivado

By checking "I AGREE" below, or OTHERWISE ACCESSING, DOWNLOADING, INSTALLING or the agreement, which can be viewed by [clicking here](#).

I Agree

3

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.2 - 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\2023.2
- C:\Xilinx\Vitis_HLS\2023.2

Download location

C:\Xilinx\Downloads\Vivado_2023.2

Disk Space Required

Download Size:	14.31 GB
Disk Space Required:	49.41 GB
Final Disk Usage:	30.37 GB
Disk Space Available:	301.79 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

4

AMD Unified Installer for FPGAs & Adaptive SoCs 2024.1 - 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\2024.1
- C:\Xilinx\Vitis_HLS\2024.1

Download location

C:\Xilinx\Downloads\Vivado_2024.1

Disk Space Required

Download Size:	13.2 GB
Disk Space Required:	47.81 GB
Final Disk Usage:	28.45 GB
Disk Space Available:	203.16 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

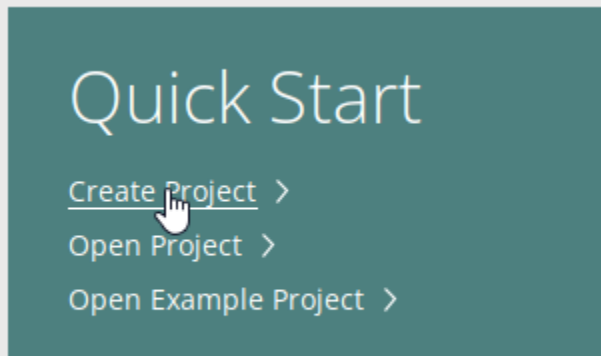
All users



Vivado

Création d'un projet

1



Create a New Vivado Project

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

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

2

New Project

3

Project Name

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

Project name:

Project location:

Create project subdirectory

Project will be created at: C:/PolyGcyLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado

< Back

Next >

< Back

Next >

New Project

4

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.

Vivado

Création d'un projet (suite)

5

New Project

Default Part

Choose a default AMD part or board for your project.

Parts | **Boards**

To fetch the latest available boards from git repository, click on 'Refresh' button. [Dismiss](#)

[Reset All Filters](#)

Vendor:

Refresh

Status

Zynq UltraScale+ PYNQ-ZU Development Board

6

New Project

Default Part

Choose a default AMD part or board for your project.


Parts | **Boards**

To fetch the latest available boards from git repository, click on 'Refresh' button. [Dismiss](#)

[Reset All Filters](#)

Vendor: Name:

Search:

Display Name	Preview	Status	Vendor	File Version	Part	I/O Pin Count	Board Rev	Available IOBs
pynq-z2		Installed	tul.com.tw	1.0	xc7z020clg400-1	400	1.0	125

Refresh Catalog was last updated on 05/31/2024 6:08:30 AM

7

New Project Summary

A new RTL project named 'CompteurGray' will be created.

The default part and product family for the new project:
Default Board: pynq-z2
Default Part: xc7z020clg400-1
Family: Zynq-7000
Package: clg400
Speed Grade: -1

To create the project, click Finish

[Back](#) [Next >](#) **Finish**

Installer le package du board pynk-z2 :

- Choisir tul.com.tw
- Appuyer sur refresh
- Un composant apparaîtra (Zynq UltraScale+PynqzU Development Board)
- Appuyer sur Téléchargement (flèche vers le bas dans status)

Vivado

Création d'un projet (suite)

8

Flow Navigator

- PROJECT MANAGER
 - Settings
 - Add Sources**
 - Language Templates
- 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

9

- Add or create constraints
- Add or create design sources
- Add or create simulation sources

10

Add Source Files

Look in:

- CompteurGray.cache
- CompteurGray.hw
- CompteurGray.ip_user_files
- CompteurGray.sim
- GrayCounter.vhd**
- Testbench_GrayCounter.vhd**

11

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
<input checked="" type="checkbox"/>	1	GrayCounter.vhd	xi_defaultlib	C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/CompteurGray
<input checked="" type="checkbox"/>	2	Testbench_GrayCounter.vhd	xi_defaultlib	C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/CompteurGray

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

Recurse all subdirectories and add found HDL files to your project.

Vivado Simulation

The image shows the Vivado IDE interface during a behavioral simulation. The left pane (PROJECT MANAGER) has the SIMULATION section expanded, with a red box highlighting the 'Run Behavioral Simulation' option in the dropdown menu. A red arrow points from this menu item to the 'Run Behavioral Simulation' button in the center pane. Another red arrow points from the center pane to the 'Tcl Console' at the bottom, which displays simulation logs.

The center pane shows the 'SIMULATION - Behavioral Simulation - Functional - sim_1 - Testbench_GrayCounter' window. It includes a 'Scope' table, an 'Objects' table, and a 'Tcl Console' window.

Name	Design Unit
Testbench_GrayCounter	Testbench_Gr
dut	GrayCounter(E

Name	Value
clk	1
reset	0
gray_out[3:0]	0
index	17
CLOCK_PERIOD	10000 ps
RESET_PERIOD	20000 ps
EXPECTED_SEQ[0:16][3:0]	0,1,3,2

```

Tcl Console
Time: 135 ns Iteration: 0 Process: /Testbench_GrayCounter/stimulus_process File: C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/Testbench_GrayCounter.vhd
Note: Comparing value = 12
Time: 145 ns Iteration: 0 Process: /Testbench_GrayCounter/stimulus_process File: C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/Testbench_GrayCounter.vhd
Note: Comparing value = 13
Time: 155 ns Iteration: 0 Process: /Testbench_GrayCounter/stimulus_process File: C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/Testbench_GrayCounter.vhd
Note: Comparing value = 14
Time: 165 ns Iteration: 0 Process: /Testbench_GrayCounter/stimulus_process File: C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/Testbench_GrayCounter.vhd
Note: Comparing value = 15
Time: 175 ns Iteration: 0 Process: /Testbench_GrayCounter/stimulus_process File: C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/Testbench_GrayCounter.vhd
Note: Comparing value = 16
Time: 185 ns Iteration: 0 Process: /Testbench_GrayCounter/stimulus_process File: C:/PolyGcyrLocal/GitHub/INF3500-AU24/Theme0/CompteurGray/Vivado/Testbench_GrayCounter.vhd
INFO: [USF-XS1m-96] XSim completed. Design snapshot 'Testbench_GrayCounter_behav' loaded.
INFO: [USF-XS1m-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:06 ; elapsed = 00:00:09 . Memory (MB): peak = 1500.895 ; gain = 27.852
    
```