UNIVERSITÉ TOULOUSE III PAUL SABATIER
Université de Toulouse

PROGRAMMATION VHDL

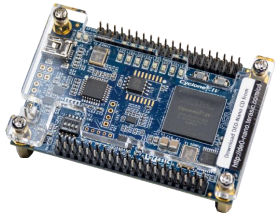
By Damian Sal y Rosas

OBJECTIF:

- Faire un premier programme VHDL sur Quartus Lite 18.1
- Prends en main le cible DE0 Nano

- Outils: **Quartus Lite**

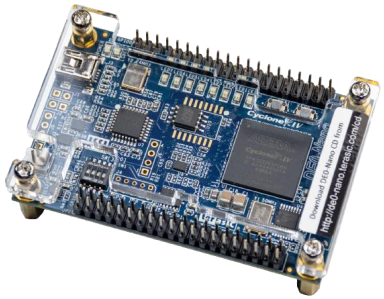
- Cible: DE0 Nano



PREMIER PROGRAMME VHDL

Visualiser le résultat sur une lumière LED

Programmation sur **Quartus Lite 18.1**



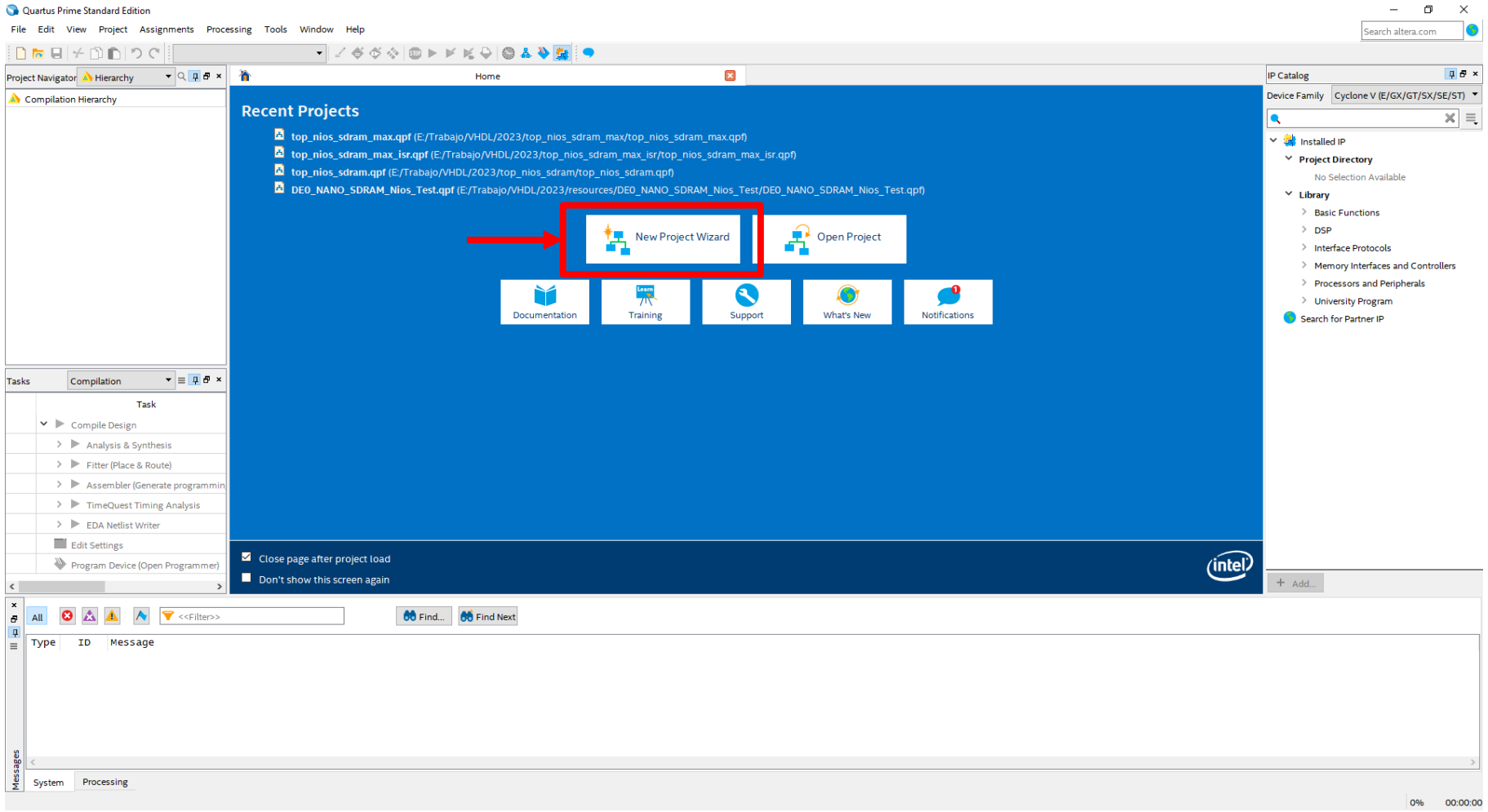
Validation sur la cible **DE0 Nano**

LET'S START …

Lancer le programme

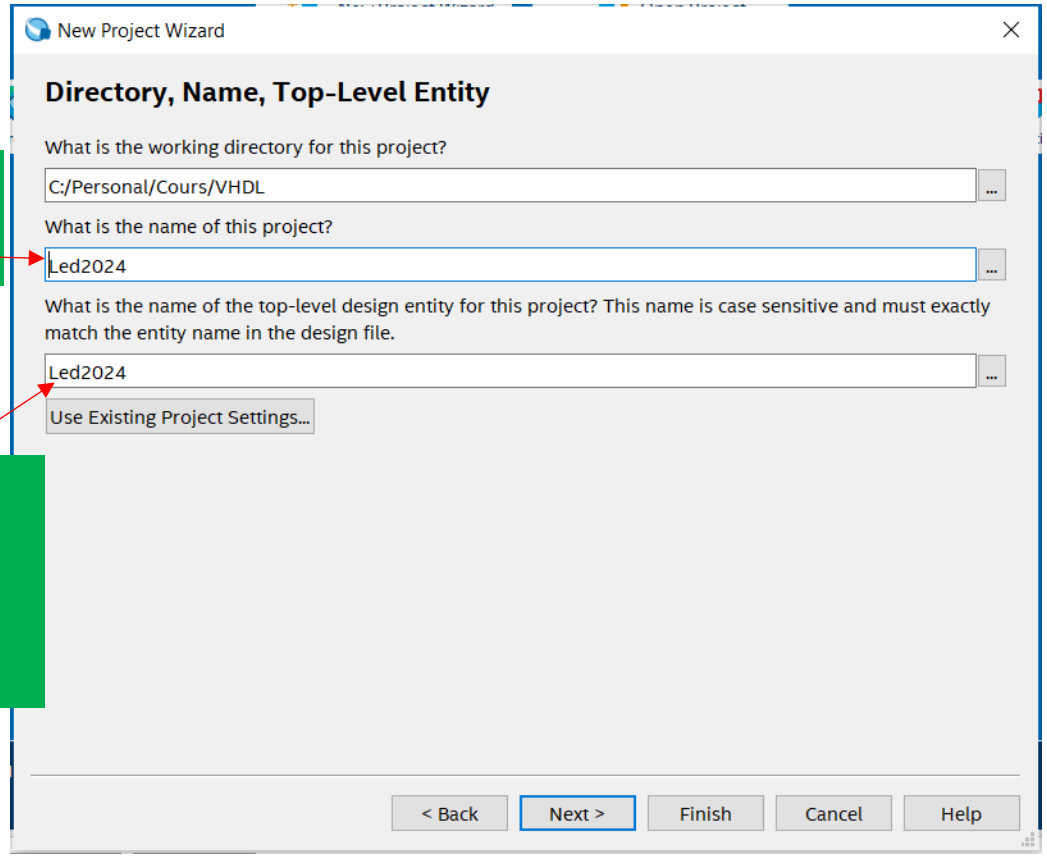
First: Il faut créer un projet

→ **New Projet Wizard**



Etape 1: Créer un fichier de travail
Etape 2: New Project Wizard → … sélectionner le directoire → Next

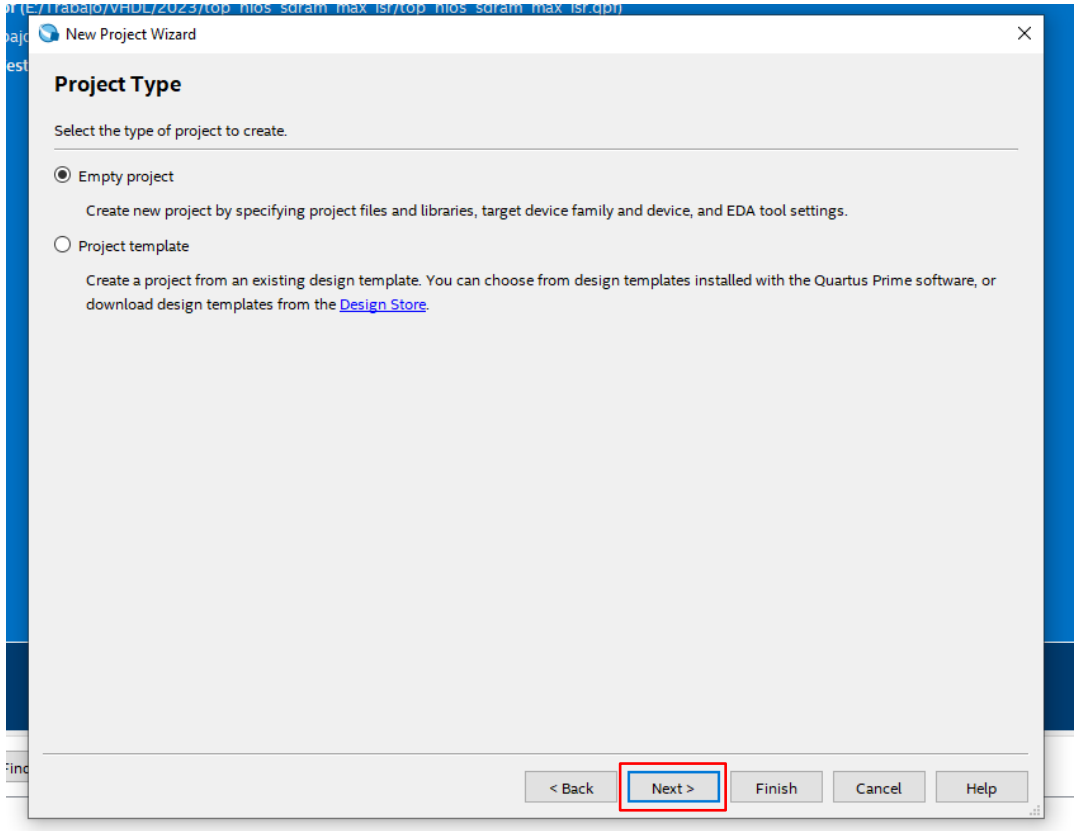
1



Nom du projet

Nom de l'Entité Top

2

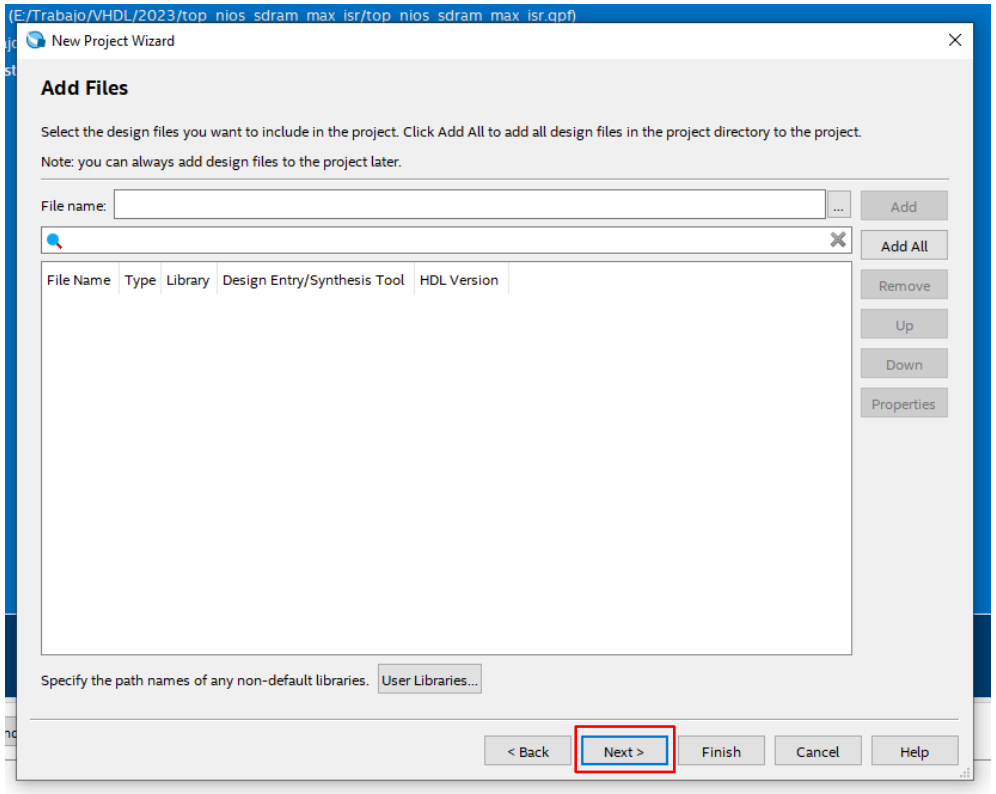


Sélectionner empty project → next

Considérations:

- **Short Name (le lien doit être court)**
- **Pas de espace dans le nom du projet**

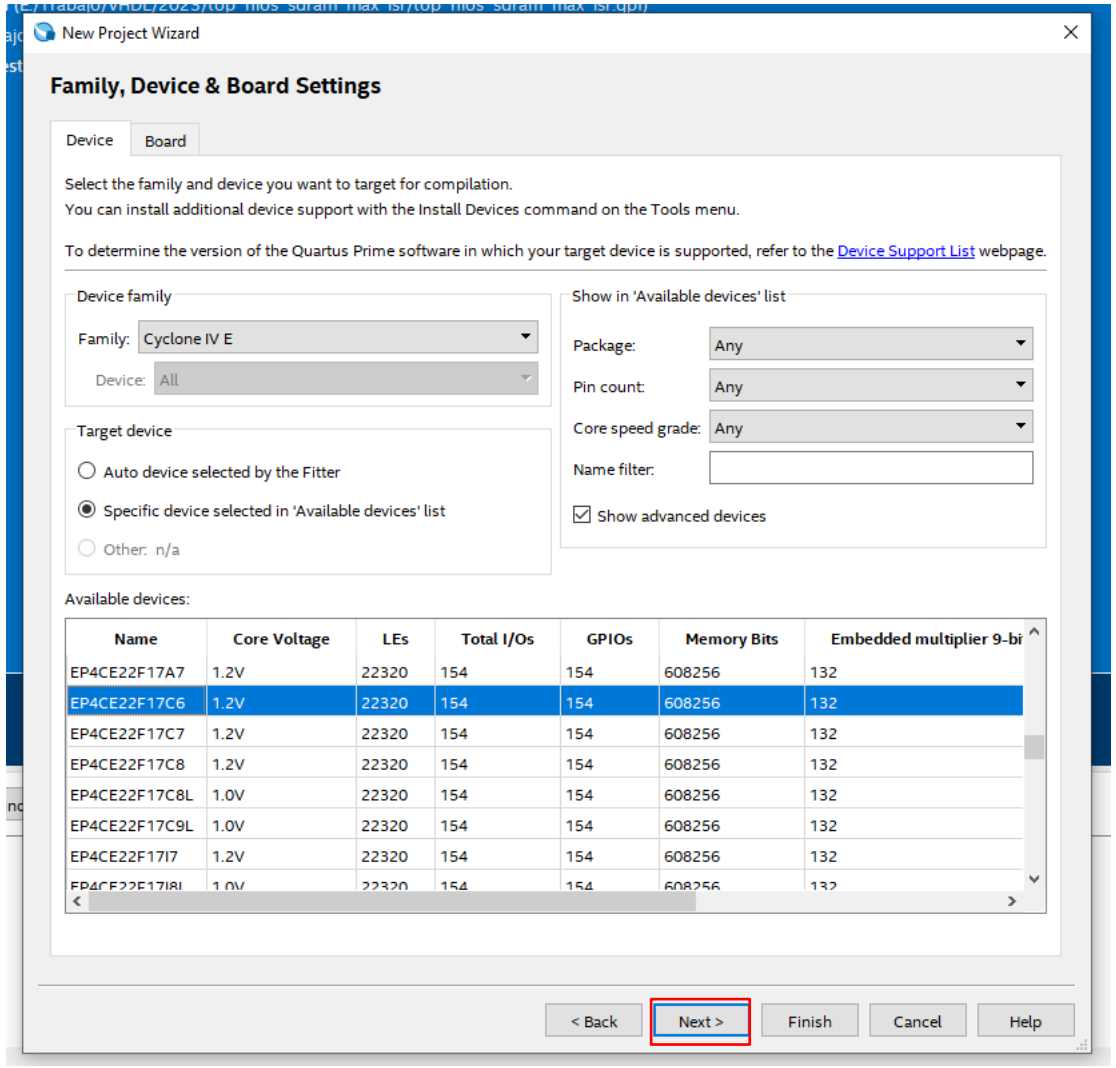
3



Etape 3:

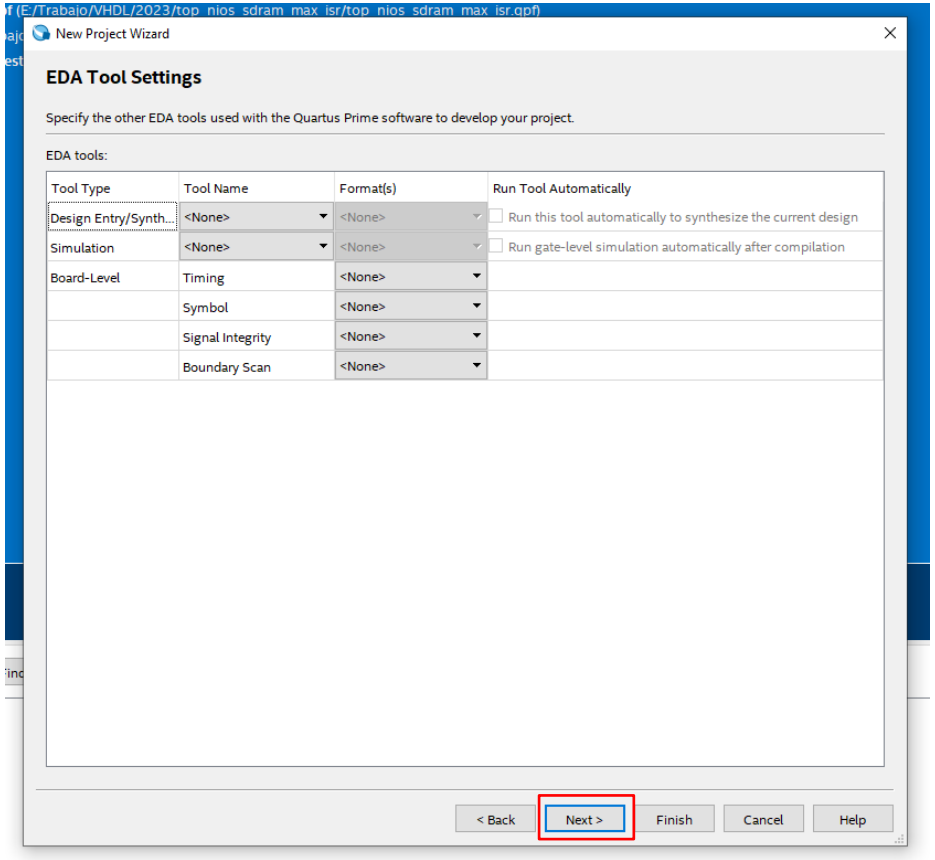
pas de fichiers à ajouter → Next

4

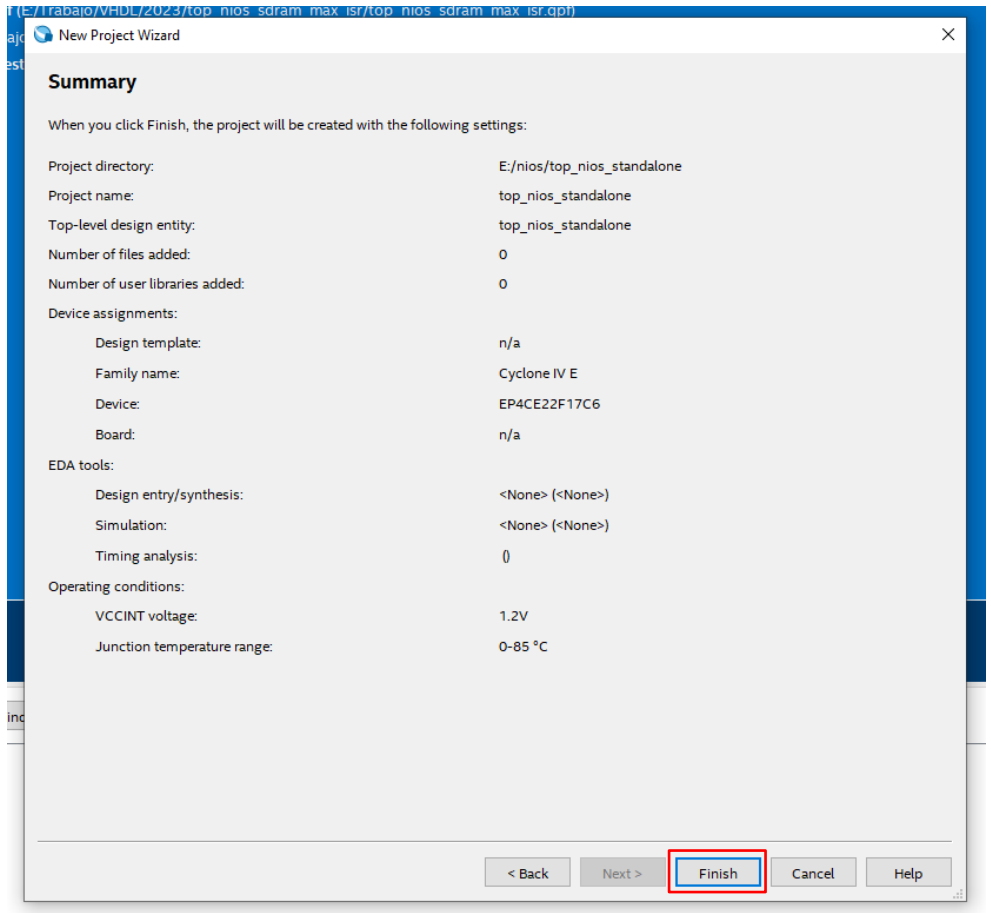


Etape 4: sélectionner FPGA . . . Altera Cyclone IV EP4CE22F17C6 → Next

5



6



Etage 5: next → finish

* Sélectionner en mode simulation Altera ModelSim – VHDL s'il faut simuler

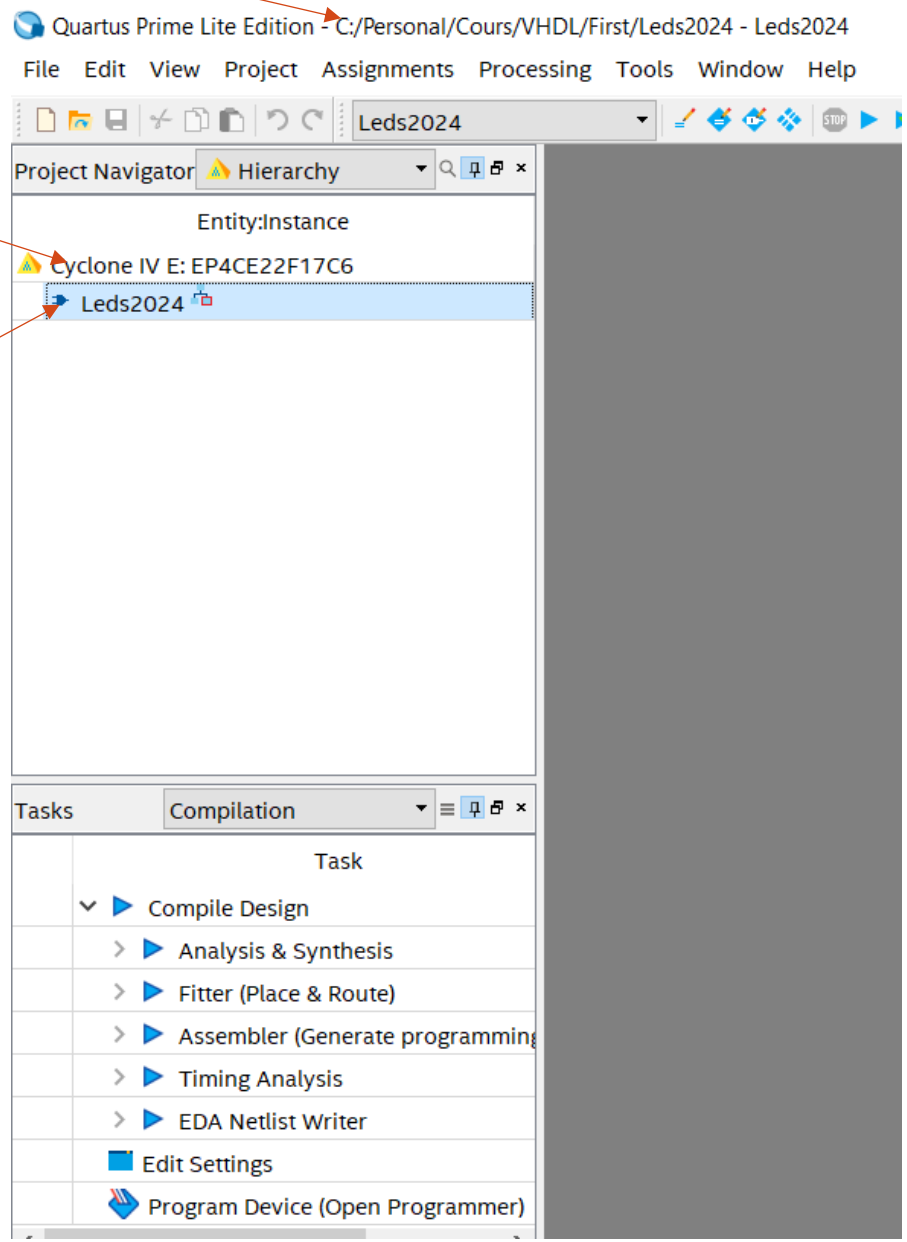
CONSOLE DE TRAVAIL

Directoire

Code du FPGA

Entité Top

Il faut ajouter le fichier VHDL pour le design du circuit



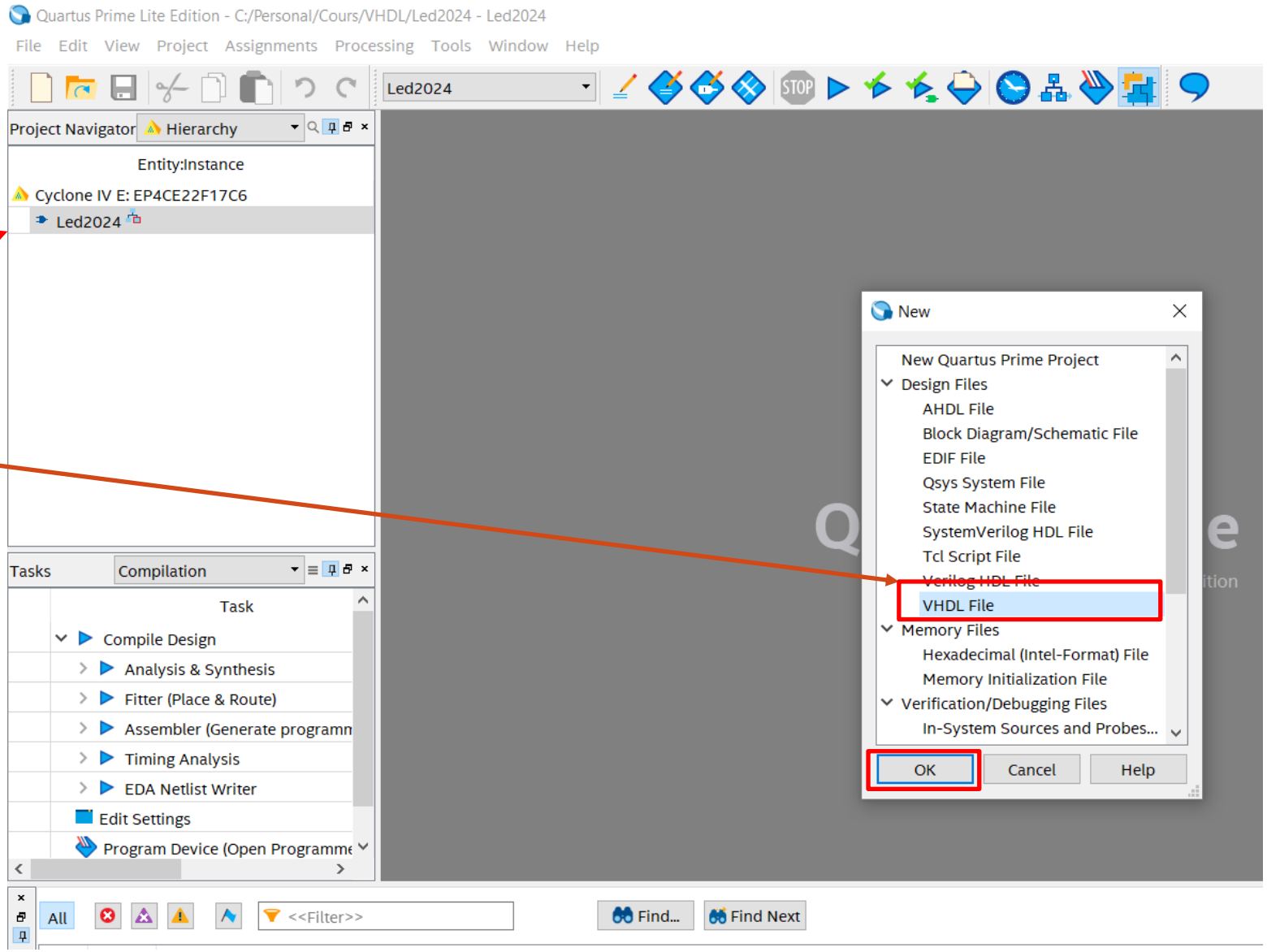
Ajouter un fichier VHDL sur la console de travail

Entité Top

Etape 1:

Sélectionner: File → New

- Choisir VHDL File -> OK



Ajouter un fichier VHDL sur la console de travail

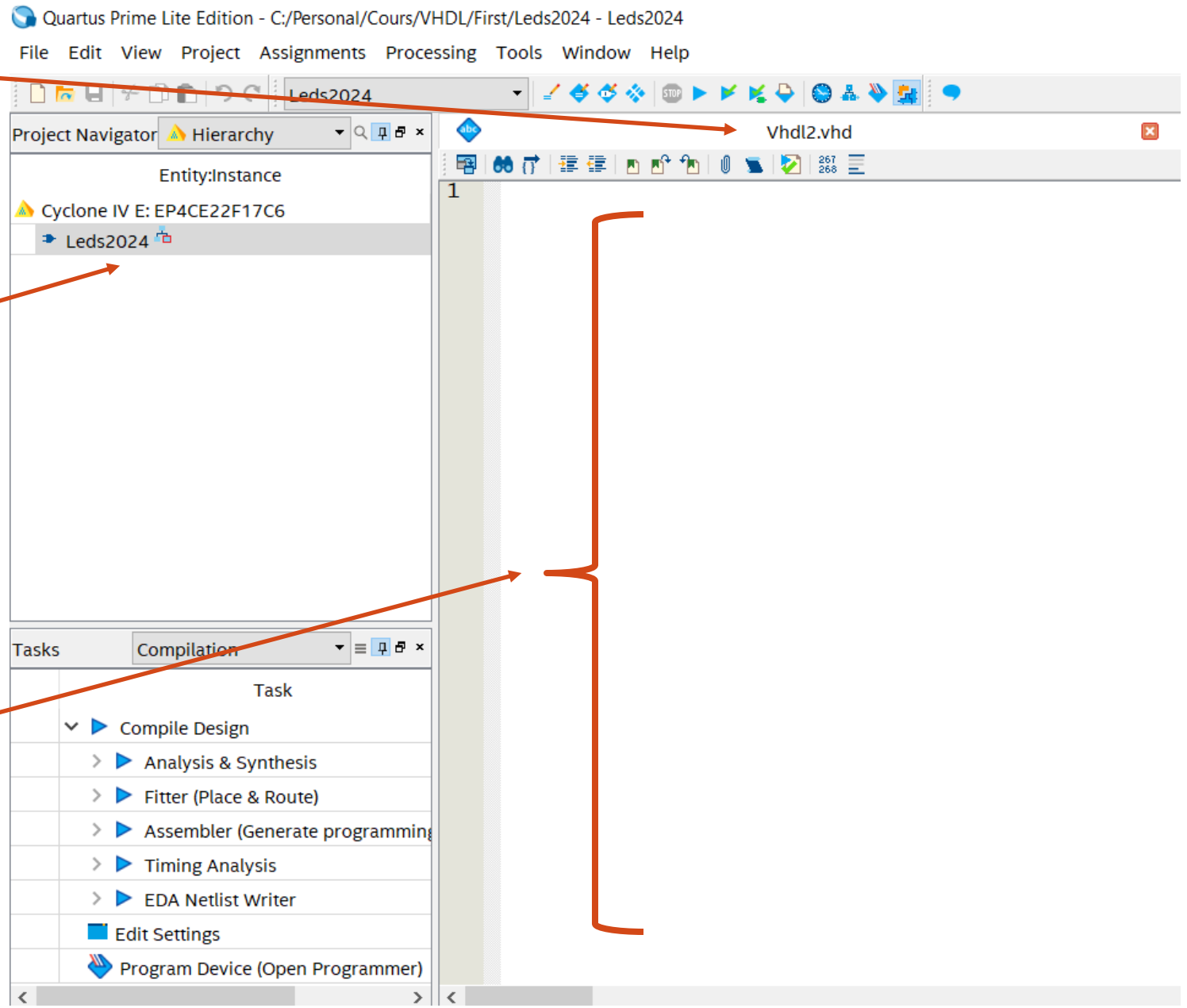
Apparait un nom par défaut:

1. Il faut modifier le nom correspondant au nom de l'ENTITE

Entité Top

Après:
Il faut ajouter le code VHDL:

- library
- Entity
- architecture



Ajouter un fichier VHDL sur la console de travail

Etape 2:

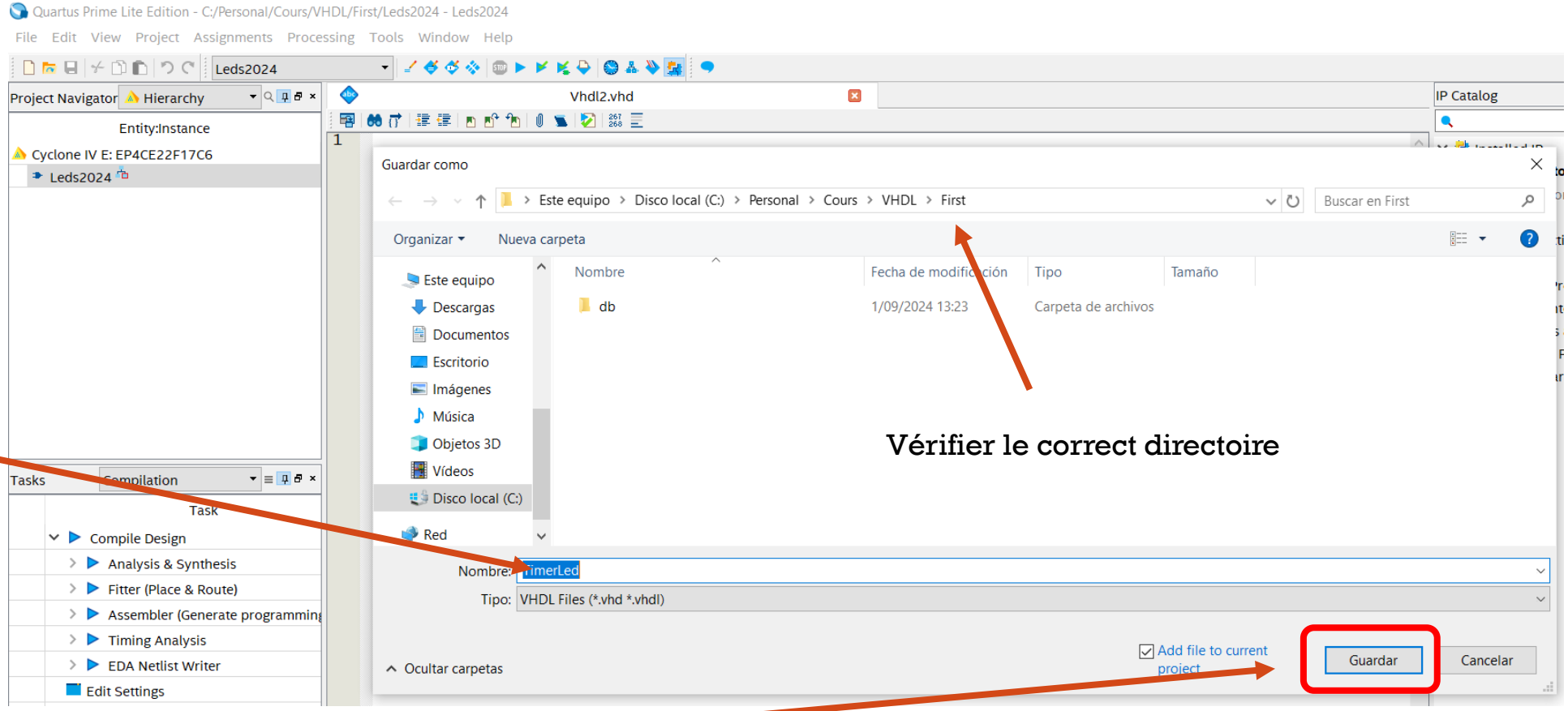
Garder le fichier avec le nom de l'entité

Exemple: TimerLed

File->SaveAs..

Indiquer le nom de l'entité

TimerLed



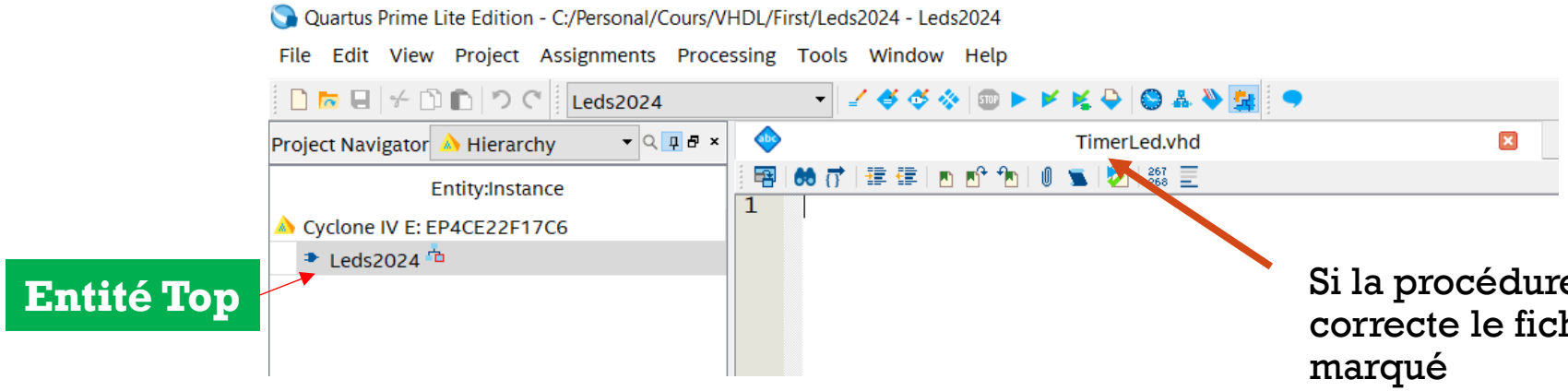
Vérifier le correct directoire

Finalement:

→ Save

Attention: l'entité s'appelle « TimerLed » et doit être désignée comme entité TOP

Ajouter un fichier VHDL sur la console de travail



Si la procédure est correcte le fichier est marqué **TimerLed.vhd**

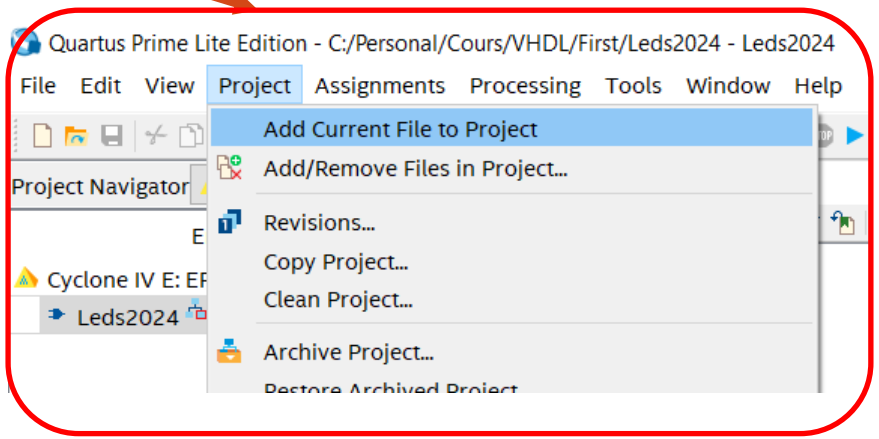
Etape 3:

On doit ajouter le fichier **TimerLed.vhd** au projet:

→Projet

→ Add current File to Projet

Attention: l'entité s'appelle « TimerLed » et doit être désignée comme entité TOP

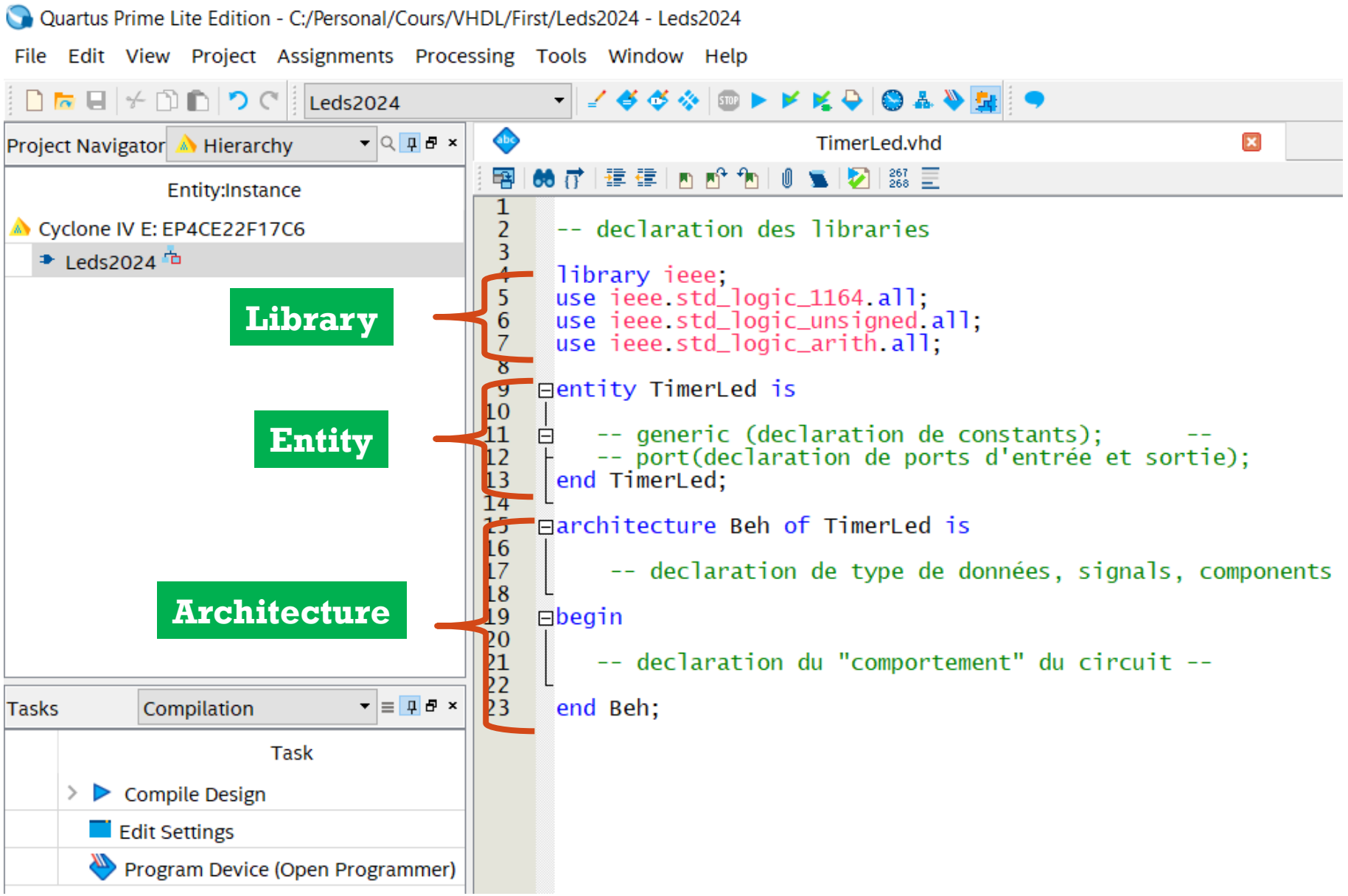


Ajouter le code « VHDL »

Etape 4:

Déclarer le code VHDL

- Library
- Entity
- Architecture



Ajouter le code « VHDL »

Erreur 2: changer l'entité « Leds2024 » à l'entité « TimerLed » comme entité TOP

Etape 5:

Compléter:

- Library
- Entity
- Architecture

Etape 6:

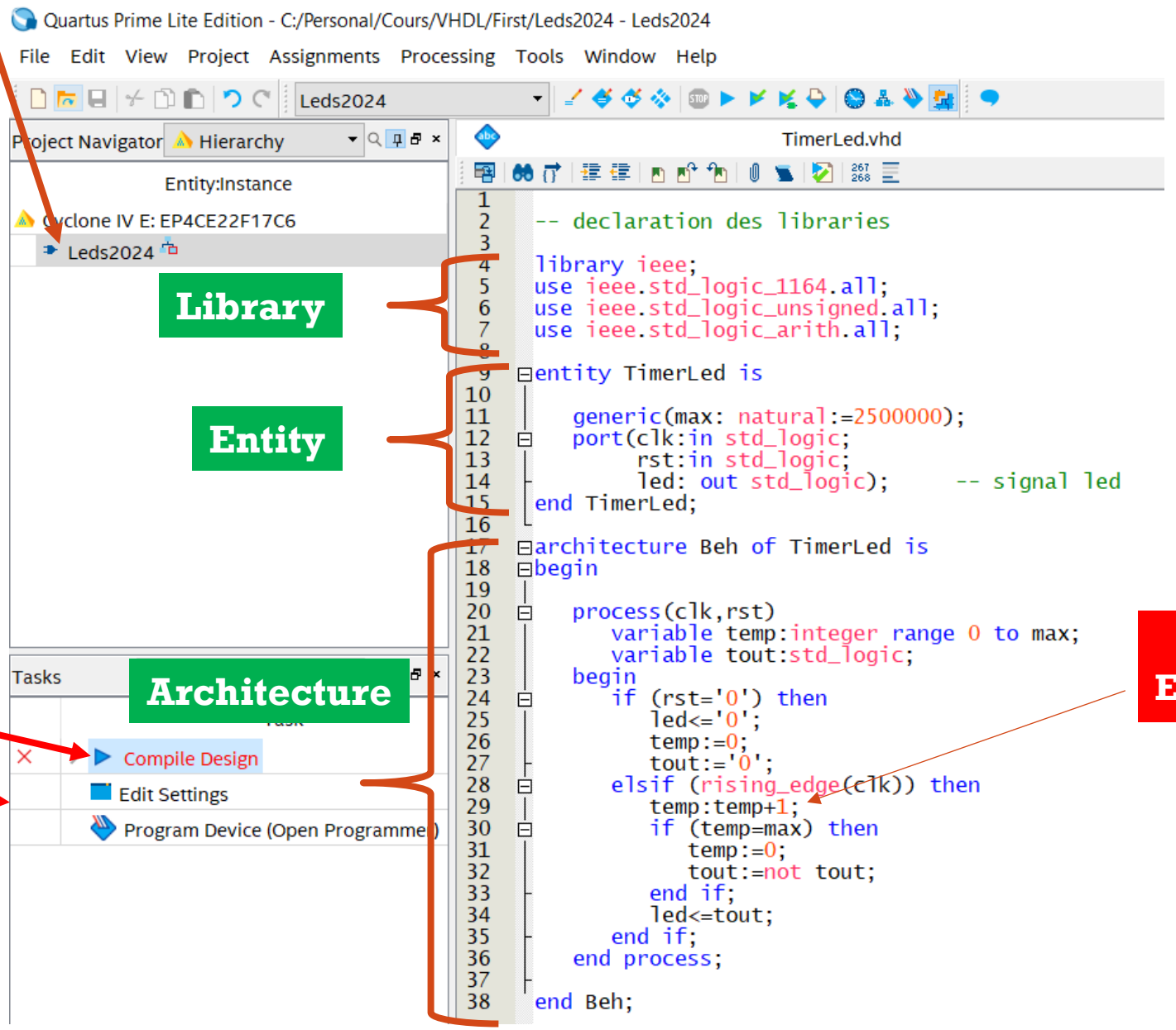
1. Clik dans:

→Compile design

2. Résultat **X**: « erreurs dans le fichier»

Erreur 1: assignation d'une variable

Erreur 2: Entité top incorrecte



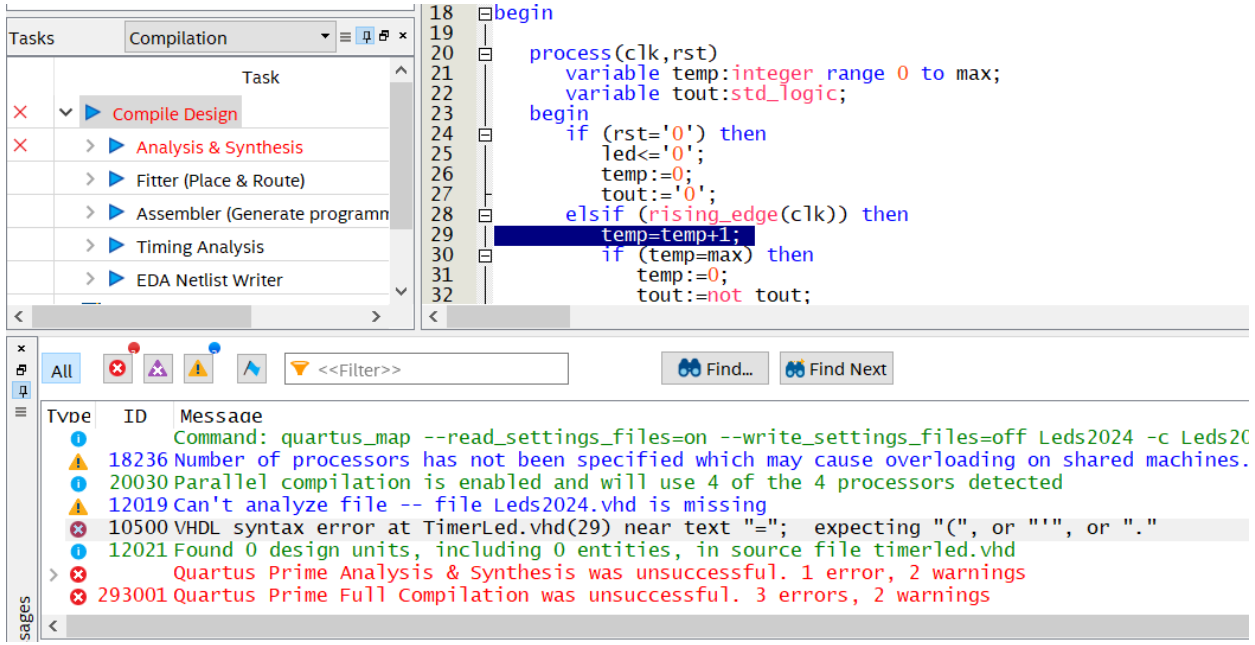
Correction du code « VHDL »

Etape 7:

Erreur 1:

Correction du code VHDL

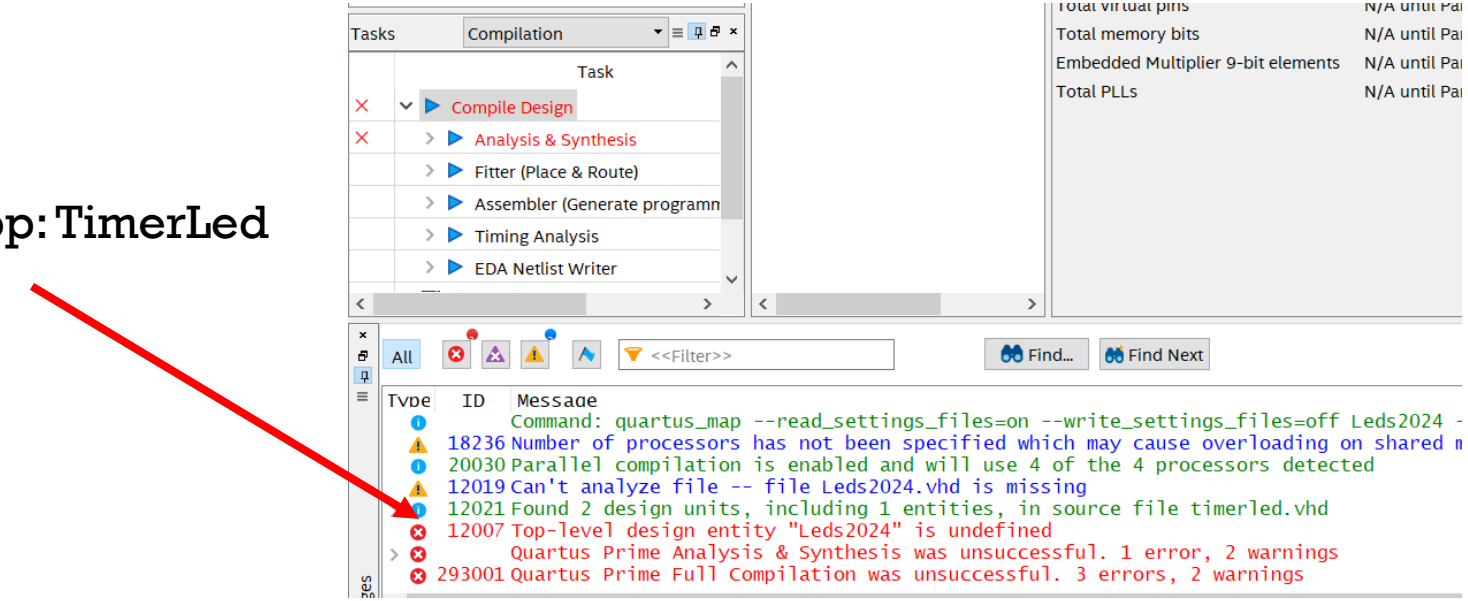
temp:=temp+1;



Etape 7:

Erreur 2:

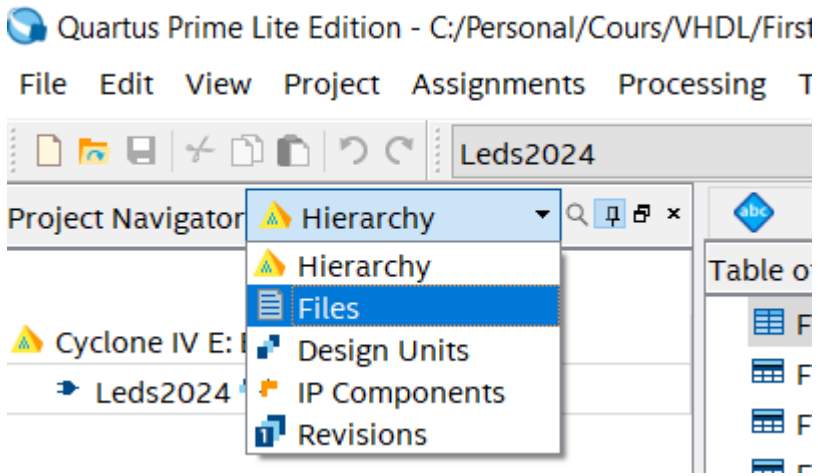
Indiquer l'entité top: TimerLed



Correction du code « VHDL »

Etape 7: Correction de l'Erreur 2:

1. Click Hierarchy → Files



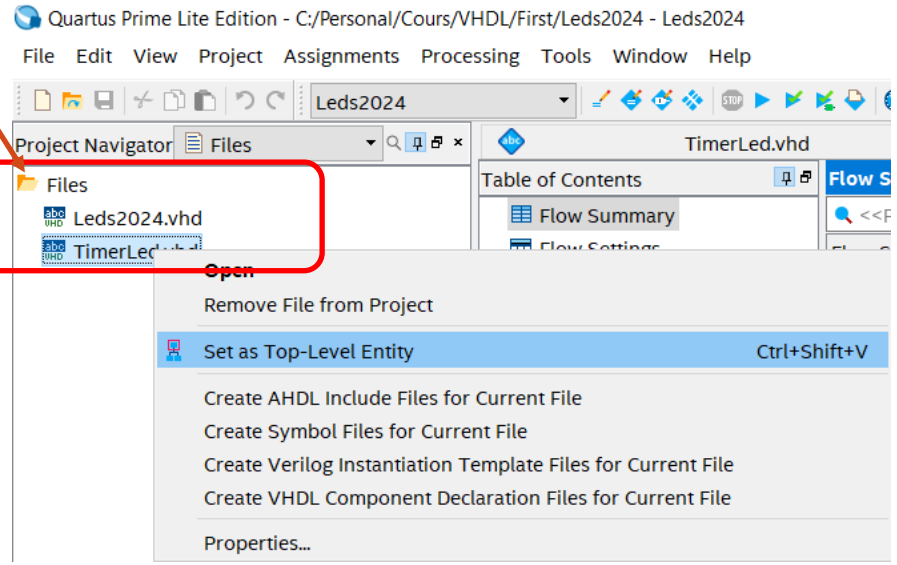
Etape 7: Correction de l'Erreur 2:

La liste des entités VHDL sont visualisées

2. Click droite in Entité TimerLet

Selectionner:

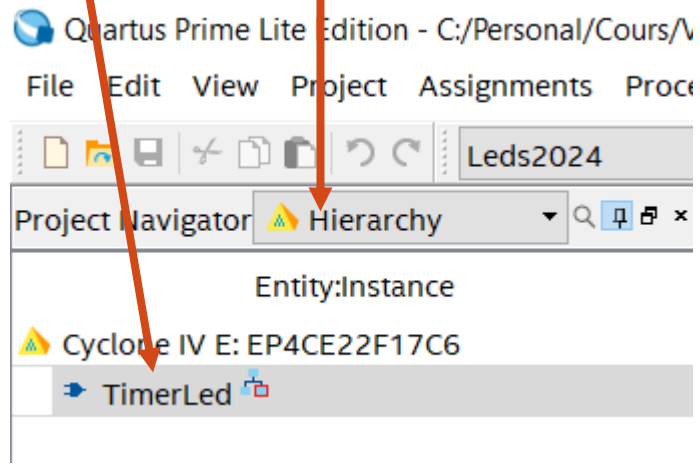
→ Set as Top-Level Entity



Etape 7: Correction de l'Erreur 2:

3. Sélectionner Hierarchy

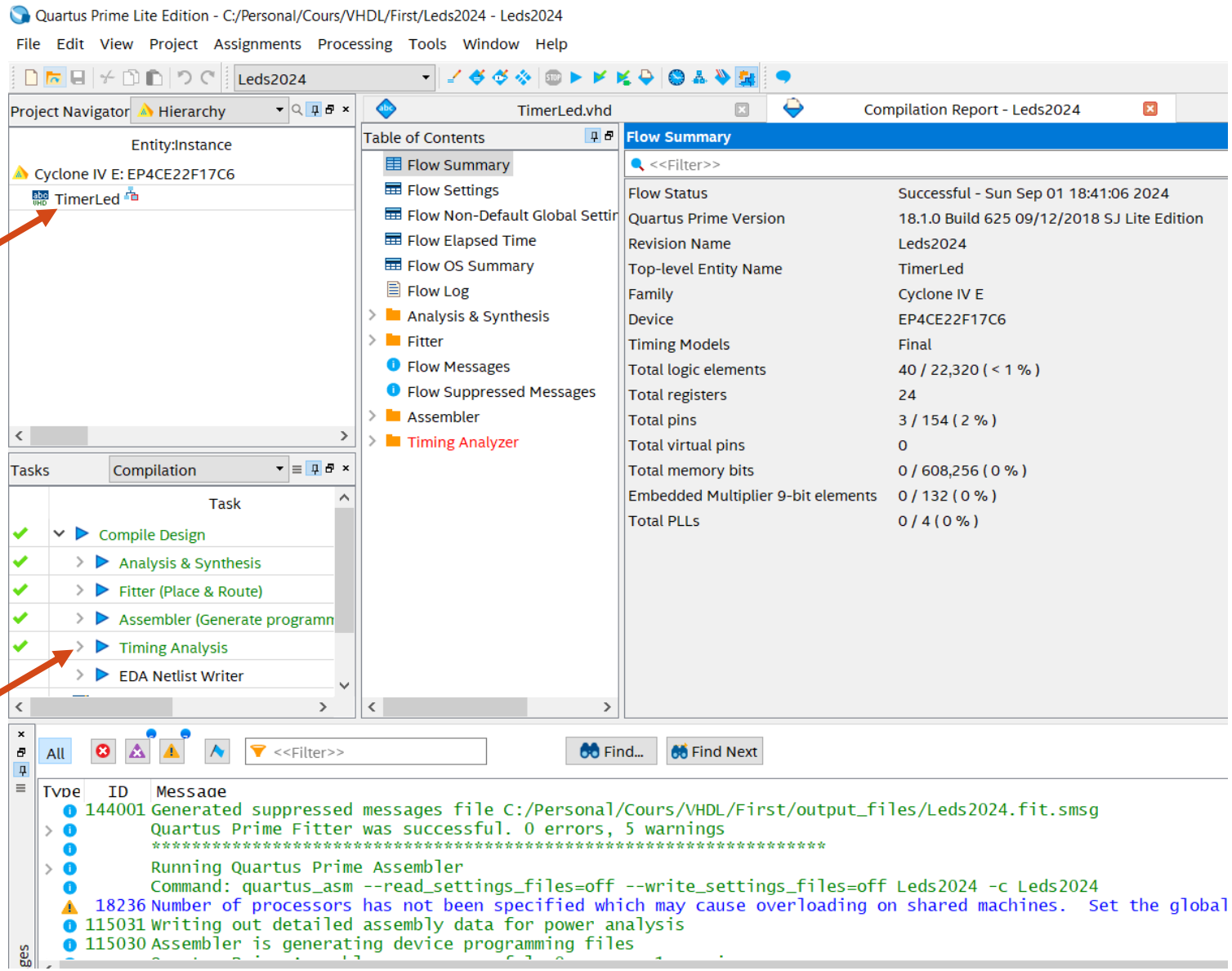
4. Vérifier que TimerLed est l'entité top



Compilation réussie du programme VHDL

Entité top:
TimerLed.vhd

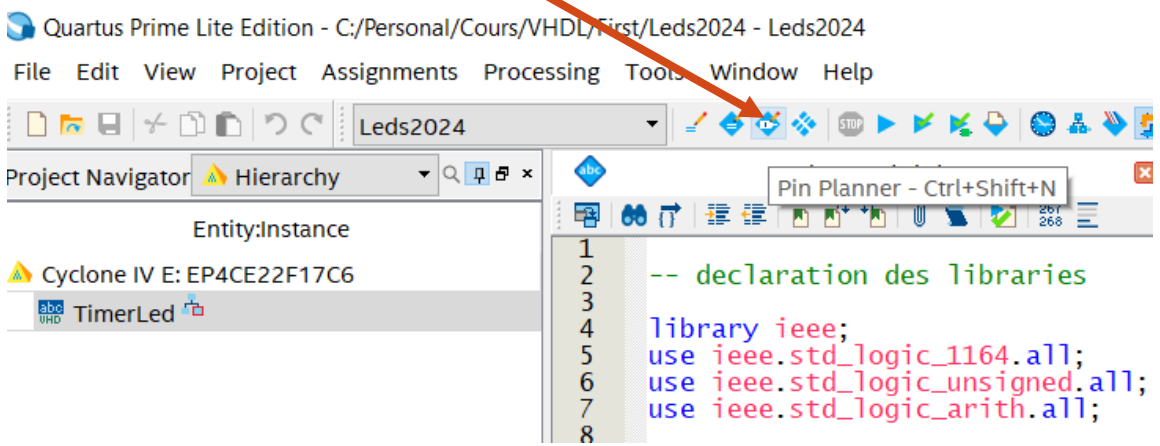
Process de compilation réussi

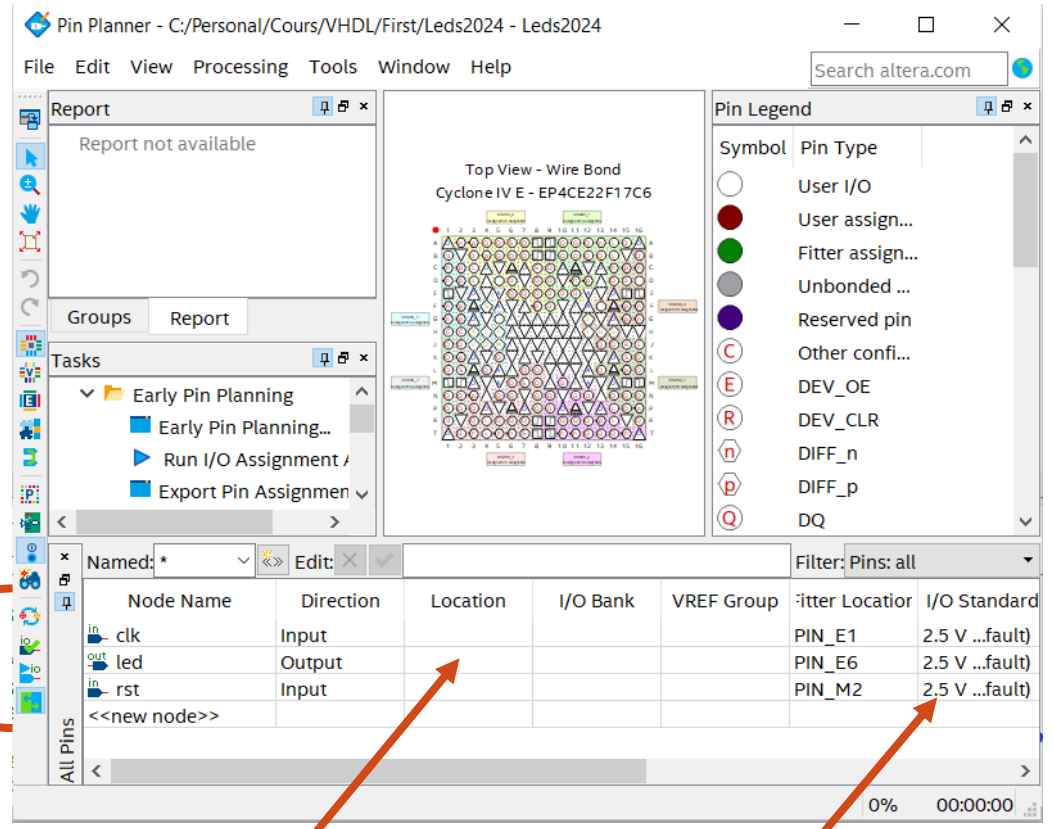


Assignation de pins

- Une fois que la compilation est réussie, il faut assigner les entrées et sorties du FPGA (pins)
- L'assignation de pins est faite avec l'option **Pin Planner**

Click → PinPlanner





Entrées et sorties de l'entité

Indiquer le port du FPGA

(voir datasheet DE0 Nano)

Indiquer la tension du port

(voir datasheet DE0 Nano)

Assignation de pins

Entrées et sorties du FPGA

rst

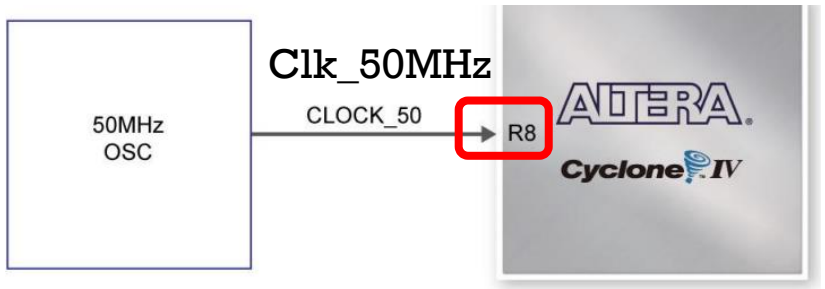
Table 3-1 Pin Assignments for Push-buttons

Signal Name	FPGA Pin No.	Description	I/O Standard
KEY[0]	PIN_J15	Push-button[0]	3.3V
KEY[1]	PIN_E1	Push-button[1]	3.3V

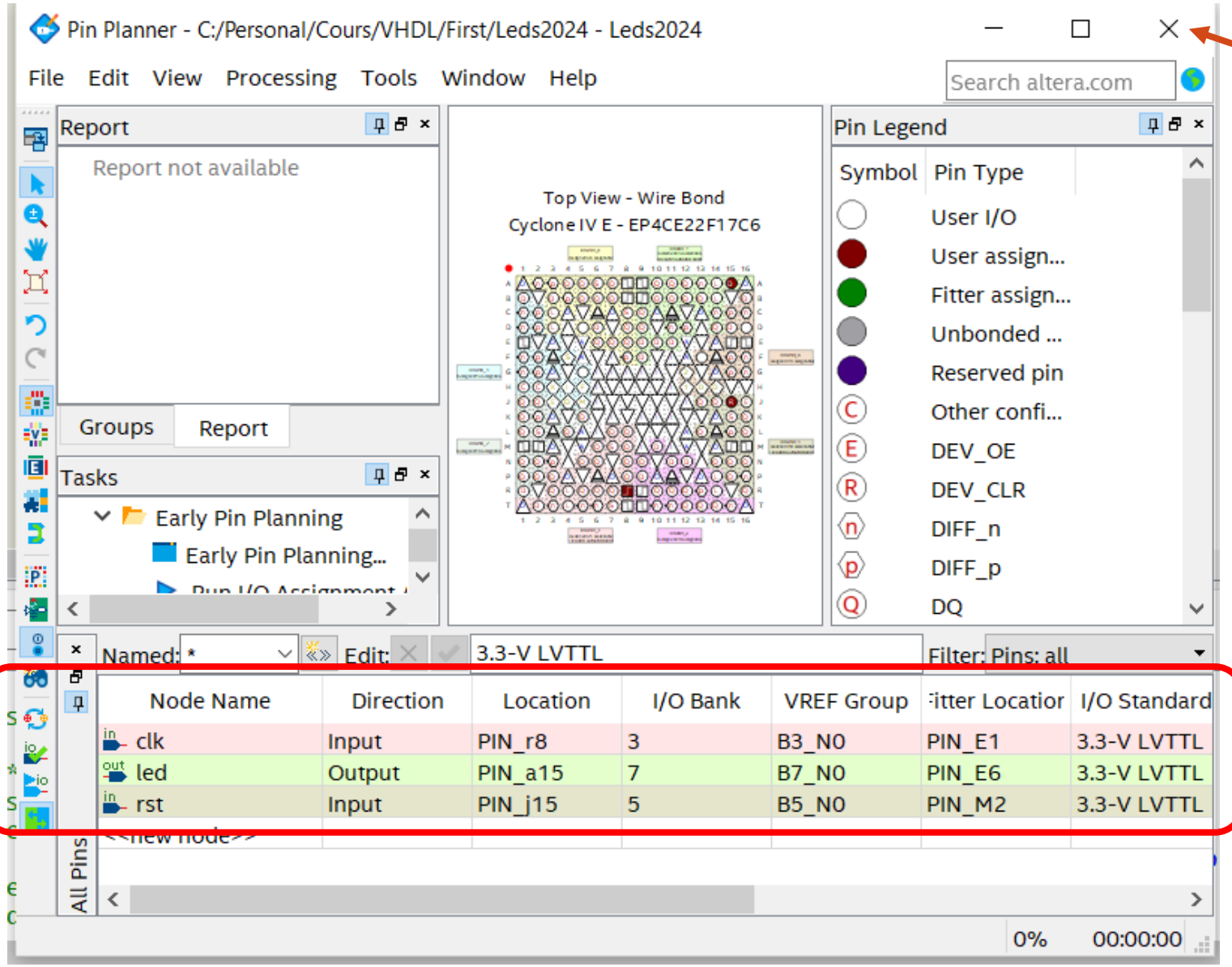
led

Table 3-2 Pin Assignments for LEDs

Signal Name	FPGA Pin No.	Description	I/O Standard
LED[0]	PIN_A15	LED Green[0]	3.3V
LED[1]	PIN_A13	LED Green[1]	3.3V
LED[2]	PIN_B13	LED Green[2]	3.3V
LED[3]	PIN_A11	LED Green[3]	3.3V
LED[4]	PIN_D1	LED Green[4]	3.3V
LED[5]	PIN_F3	LED Green[5]	3.3V
LED[6]	PIN_B1	LED Green[6]	3.3V
LED[7]	PIN_L3	LED Green[7]	3.3V



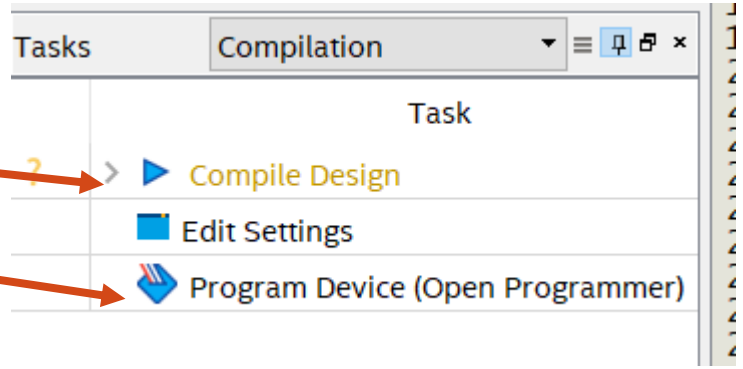
Assignation de pins

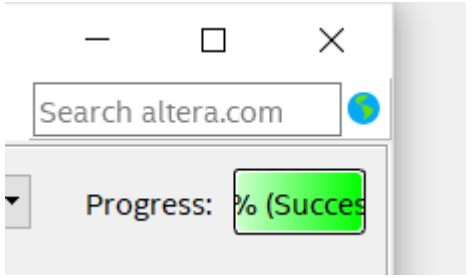


Fermer la fenêtre après l'assignation de pins

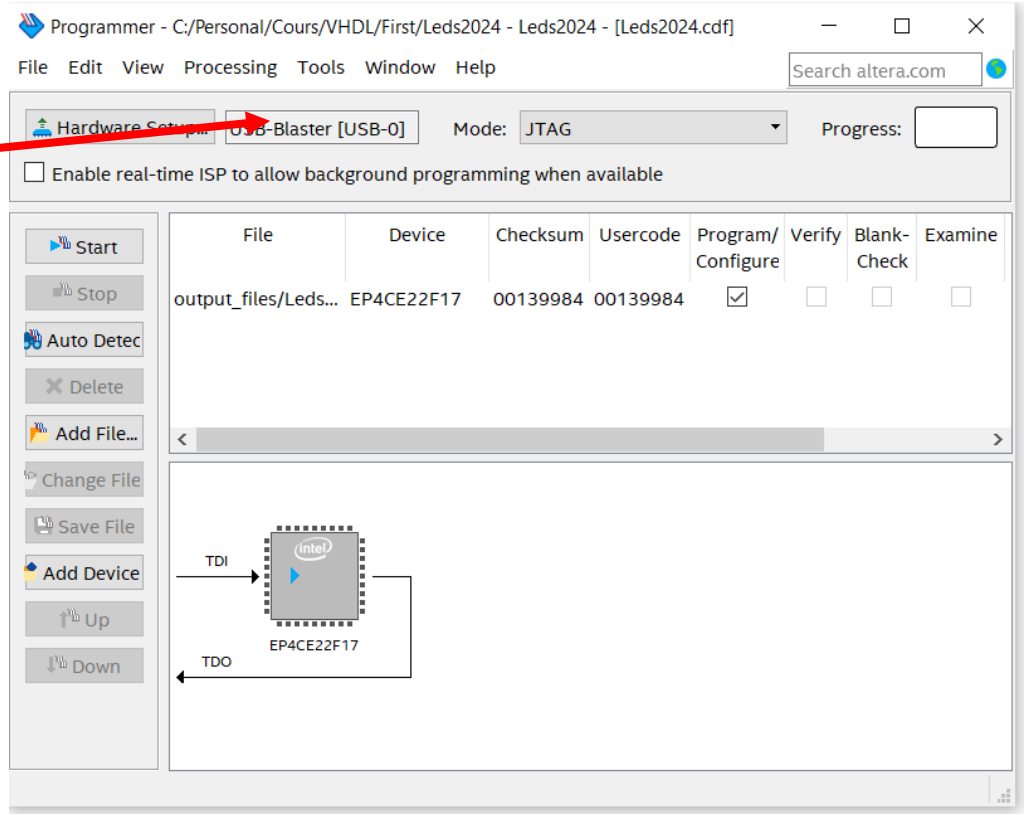
Validation sur la carte

1. Compiler une 2eme fois (click)
2. Programmer la carte





3. Sélectionner USB-Blaster
4. Envoi du fichier **.sof** vers la carte



5. Programmation ok