

*TP N°1\_Part1*  
*Prise en main de Quartus II V13*  
*Introduction à la synthèse logique*  
*Programmation d'un FPGA Cyclone II*  
*sur carte DE2*  
*Durée: 2h max*



**Préambule**

Au cours de ce TP vous allez découvrir l'environnement de CAO d'ALTERA/INTEL, la suite QUARTUS II\_V13 Sp1 avec licence.

Une carte DE2, de développement est mise à votre disposition pour mettre en œuvre les concepts de base du langage de description VHDL. Le matériel est **fragile** (décharges électrostatiques, connectique, USB, etc.). **Prenez-en soin !**

Le non-respect des consignes d'utilisation de ce matériel peut entraîner **sa destruction** et aura comme conséquence l'attribution de la note **ZERO** pour cette partie de l'UE.

Il y aura d'autres précautions à prendre par la suite pour ne pas détruire le FPGA de la carte DE2.

Le PC sur lequel vous allez développer, tourne en dual boot Linux Ubuntu et Windows 7. Pour ce premier TP nous travaillerons sous Windows 7.

Un dossier **Ressources\_DE2** disponible sur l'ENT contient tous les documents nécessaires pour réaliser ce TP.

Il faudra créer sur le bureau un dossier de travail identifié ainsi :

**TP\_QuartusII\_V13**

Dans ce dossier il y aura un sous-dossier **TP1\_QII\_VotreNom** qui contiendra tous les fichiers élaborés au cours du **TP1\_Part1**. Il contiendra en particulier le nom du projet :

**TP1\_QII\_V13**

**Contenu du kit (à vérifier à chaque début séance !).**

Le PC et le kit de développement qui vous sont attribués **sont numérotés**. Le kit et le PC porteront nécessairement le même numéro. Vous conserverez ces mêmes numéros pour chacune des séances de TP nécessitant leur utilisation.

Vous disposez dans le kit du matériel suivant :

- Un câble USB-A-USB-B,
- Une carte « ALTERA DE2 » dans son emballage d'origine.

A la fin de chaque séance qui nécessite l'utilisation de cette carte vous la remettrez dans sa boîte. Avant de la refermer appeler votre enseignant **OBLIGATOIREMENT !**

**A NE PAS OUBLIER !**

Un **compte rendu de TP** est à déposer en temps et en heure dans l'ENT (dépôt d'un devoir) (**TP1\_Part1\_QII\_VotreNom.pdf**) ainsi que le projet complet au format zip.

## A) Lancement de QuartusII V13

**QuartusII** permet la « programmation » des circuits programmables FPGA d'ALTERA/INTEL. La « programmation » d'un circuit se décompose en 4 phases :

1. **Saisie de la description** du circuit. Pour cela 3 outils de description sont utilisables :

- + Description sous la forme d'un schéma électronique,
- + Description textuelle en utilisant un langage de « programmation » (VHDL, Verilog, AHDL, SystemC, etc.),
- + Description sous forme de machines à états (FSM).

2. **Vérification de la description** par « compilation » du circuit.

3. **Vérification du bon fonctionnement** de la description par simulation logico-fonctionnelle et logico-temporelle.

→ A ce stade, il peut être nécessaire de reprendre à la première phase,

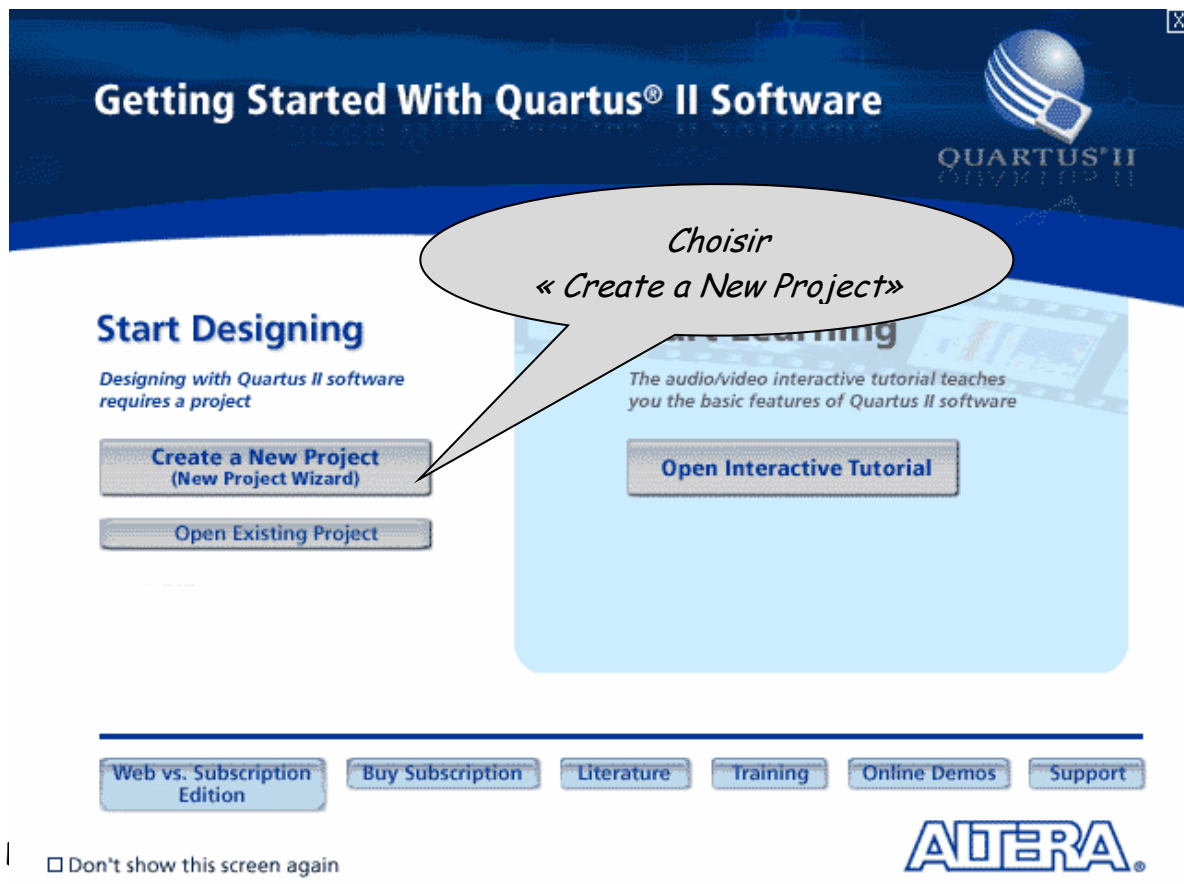
4. **Configuration (Programmation)** du circuit physique (cible matérielle).

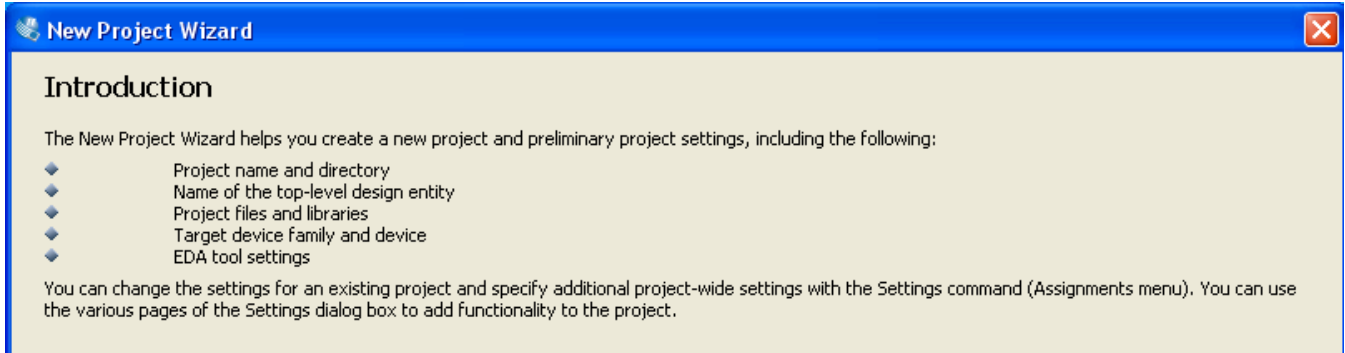
Les étapes à réaliser qui suivent sont nécessaires pour définir votre contexte de travail dans l'environnement de **Quartus II** et en particulier pour la cible matérielle (FPGA) que vous allez **configurer** (programmer).

**Quartus II** fonctionnant par projet il est nécessaire de commencer par créer un projet à chaque nouvelle conception.

+ **Étape N°0** : Lancer l'application à l'aide du raccourci présent sur le bureau.

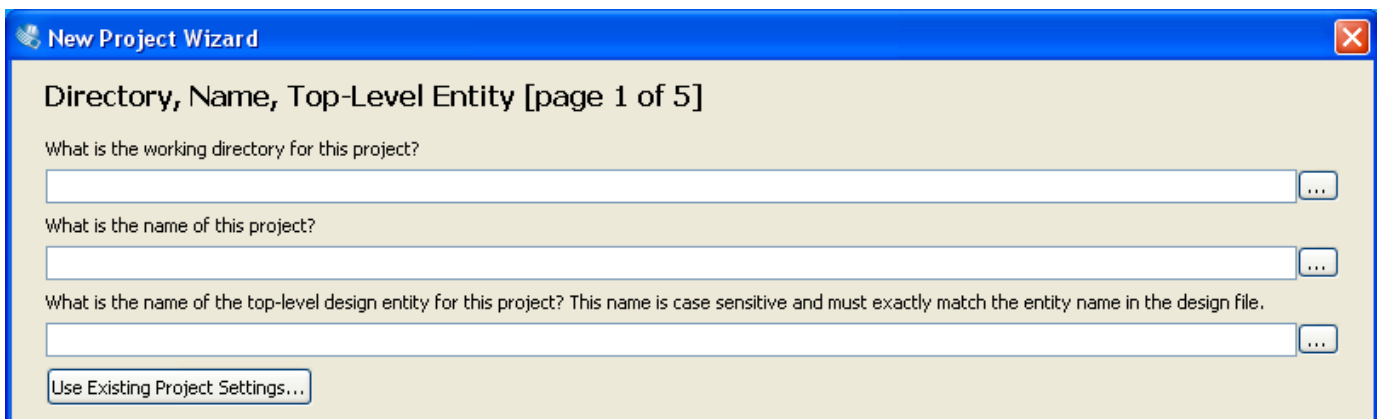
La fenêtre de l'IDE de **QuartusII** s'ouvre avec en premier plan celle ci-dessous :



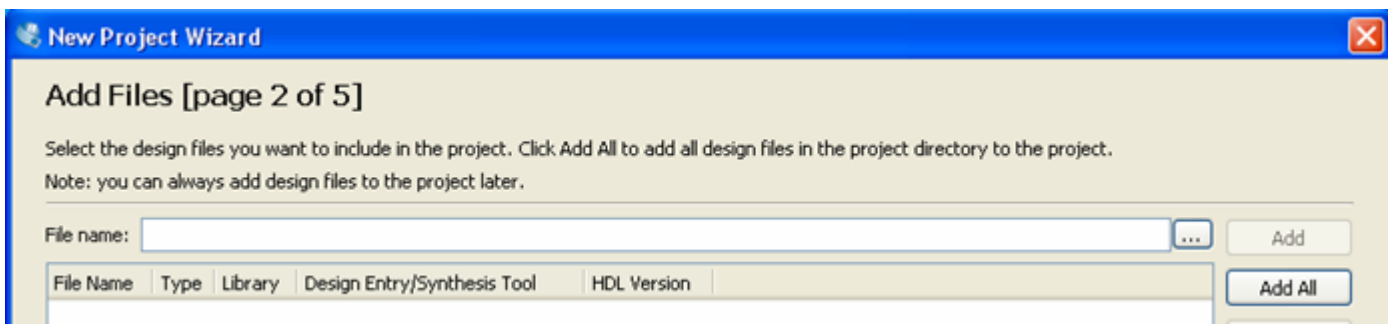


Les fenêtres suivantes vous indiquent les cinq étapes à suivre pour construire votre projet.

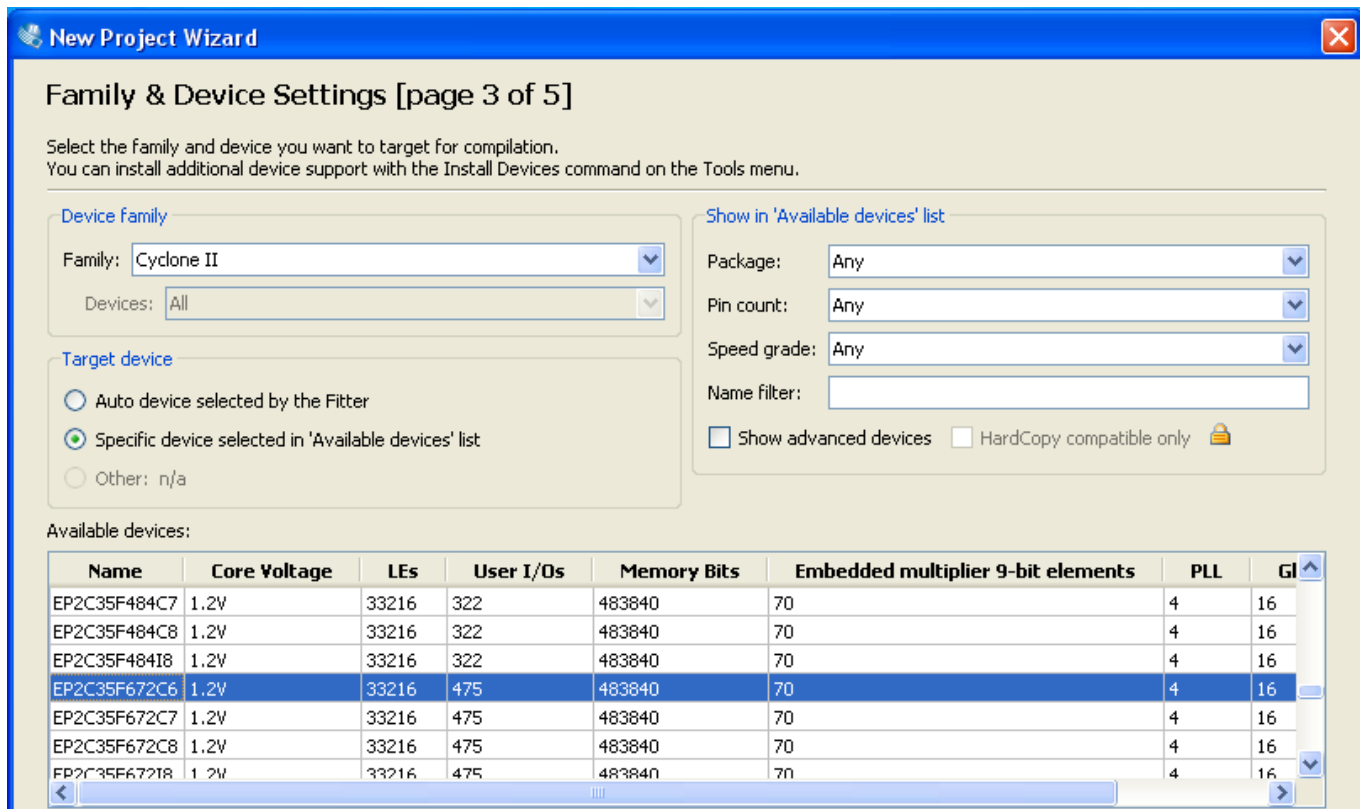
✚ **Étape N°1** : Compléter les champs conformément aux indications données en début du TP :



✚ **Étape N°2** : Cette étape ne nous concerne pas ! Cliquer sur « Next »



➤ **Étape N°3 :** Cette étape vous permet de choisir notre cible **FPGA** : **EP2C35F672C6** de la famille **Cyclone II**. Cliquer sur « Next ».



Select the family and device you want to target for compilation.  
You can install additional device support with the Install Devices command on the Tools menu.

**Device family**

Family: Cyclone II  
Devices: All

**Target device**

☐ Auto device selected by the Fitter  
☒ Specific device selected in 'Available devices' list  
☐ Other: n/a

**Show in 'Available devices' list**

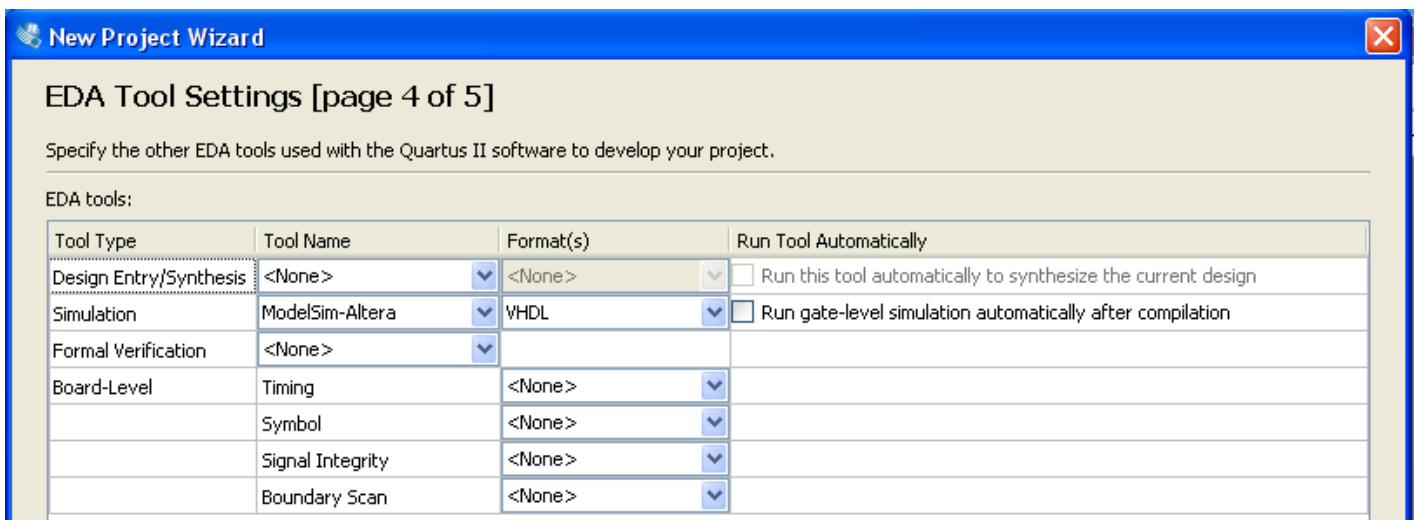
Package: Any  
Pin count: Any  
Speed grade: Any  
Name filter:   
☐ Show advanced devices ☐ HardCopy compatible only

Available devices:

Name	Core Voltage	LEs	User I/Os	Memory Bits	Embedded multiplier 9-bit elements	PLL	GL
EP2C35F484C7	1.2V	33216	322	483840	70	4	16
EP2C35F484C8	1.2V	33216	322	483840	70	4	16
EP2C35F484I8	1.2V	33216	322	483840	70	4	16
EP2C35F672C6	1.2V	33216	475	483840	70	4	16
EP2C35F672C7	1.2V	33216	475	483840	70	4	16
EP2C35F672C8	1.2V	33216	475	483840	70	4	16
EP2C35F672I8	1.2V	33216	475	483840	70	4	16

➤ **Étape N°4 :** Compléter la fenêtre comme indiqué ci-dessous. Si le timing nous le permet nous aborderons la simulation...

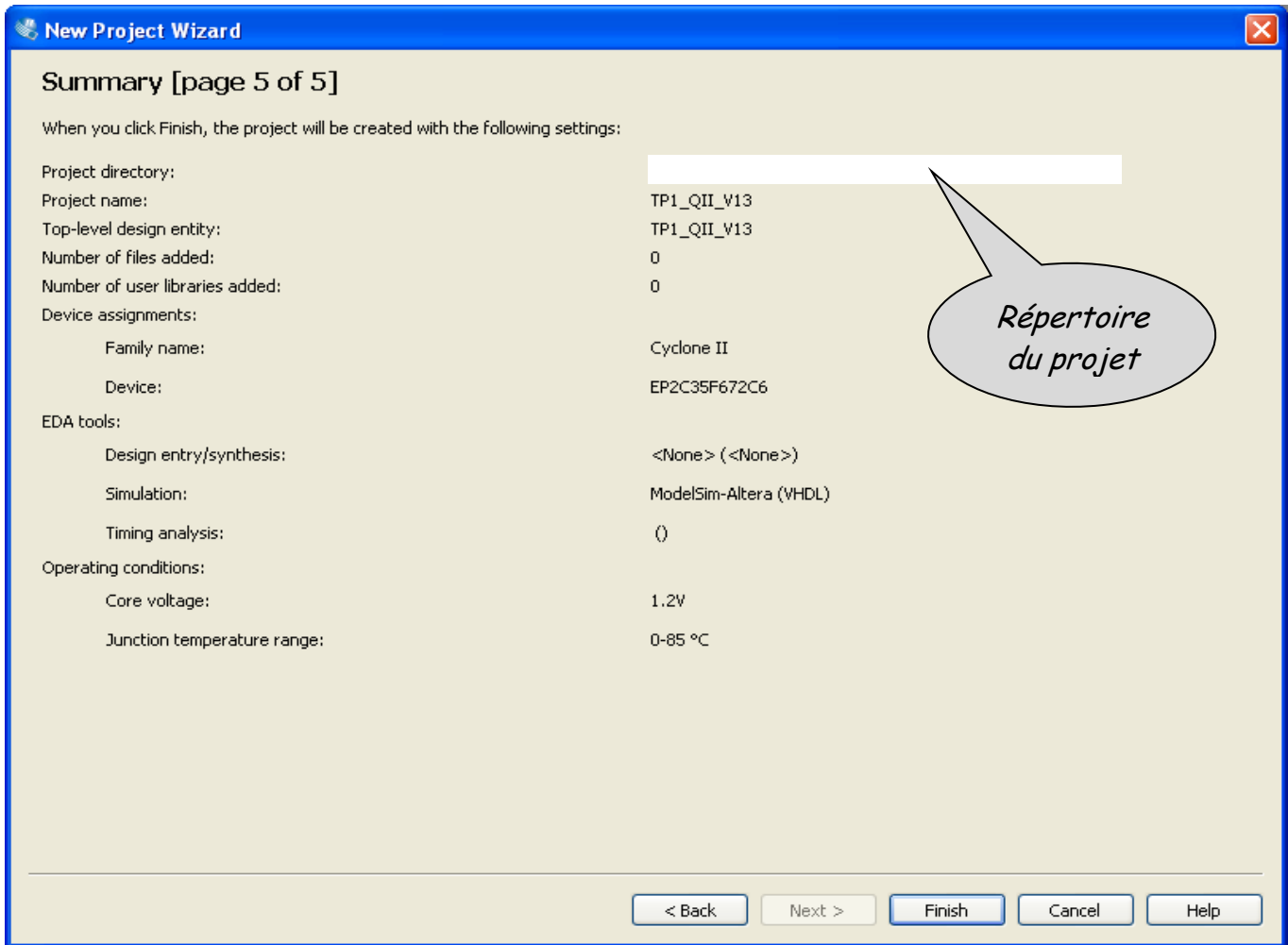
**Spécifier** le logiciel de simulation **ModelSim-Altera** depuis le menu déroulant, et assurez-vous que le format **VHDL** est sélectionné. Ne cochez pas l'option **Run gate-level simulation automatically after compilation**. Laissez les autres outils et formats à leurs valeurs par défaut (<None>);



Specify the other EDA tools used with the Quartus II software to develop your project.

EDA tools:

Tool Type	Tool Name	Format(s)	Run Tool Automatically
Design Entry/Synthesis	<None>	<None>	<input type="checkbox"/> Run this tool automatically to synthesize the current design
Simulation	ModelSim-Altera	VHDL	<input type="checkbox"/> Run gate-level simulation automatically after compilation
Formal Verification	<None>		
Board-Level	Timing	<None>	
	Symbol	<None>	
	Signal Integrity	<None>	
	Boundary Scan	<None>	



**New Project Wizard**

**Summary [page 5 of 5]**

When you click Finish, the project will be created with the following settings:

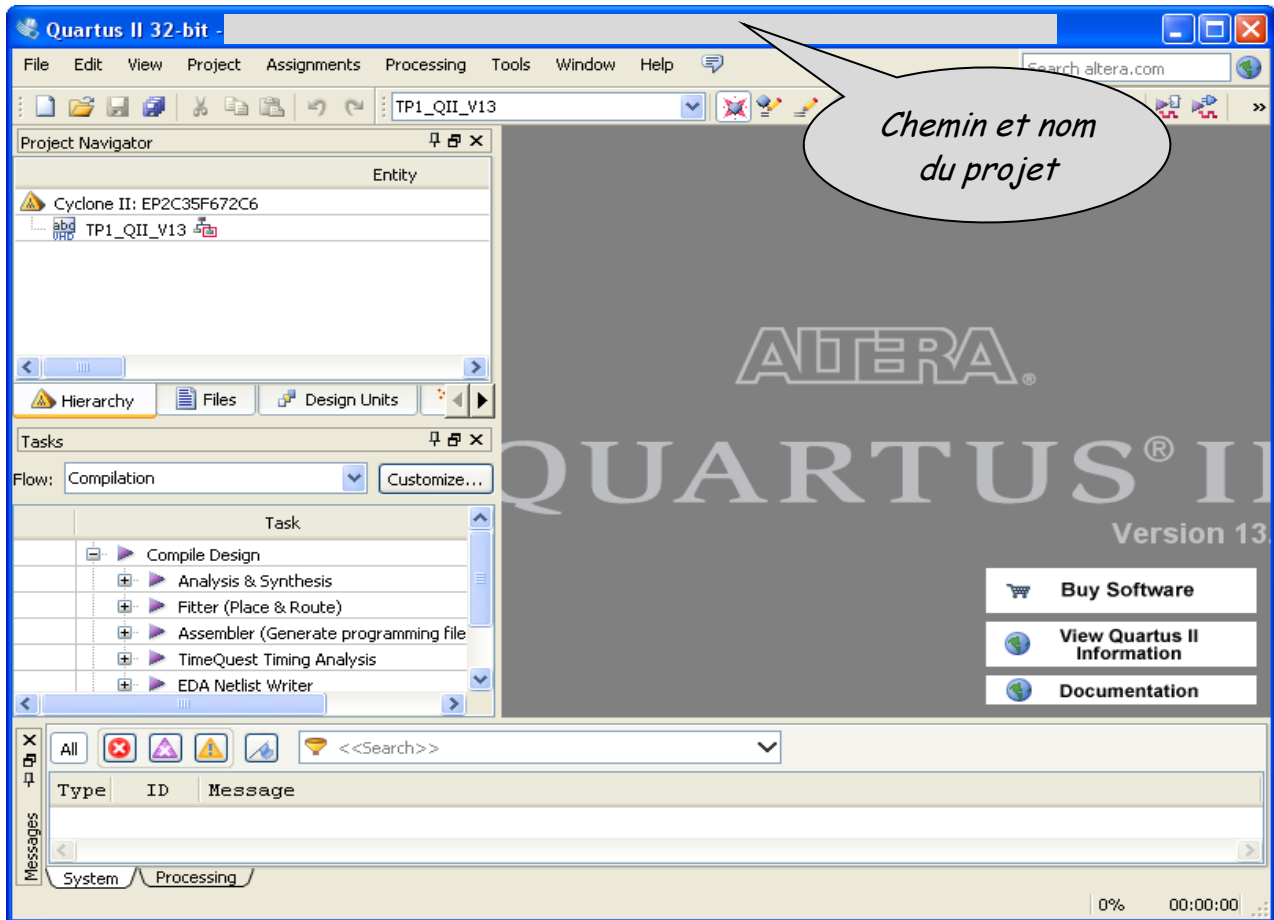
Project directory:	
Project name:	TP1_QII_V13
Top-level design entity:	TP1_QII_V13
Number of files added:	0
Number of user libraries added:	0
Device assignments:	
Family name:	Cyclone II
Device:	EP2C35F672C6
EDA tools:	
Design entry/synthesis:	<None> (<None>)
Simulation:	ModelSim-Altera (VHDL)
Timing analysis:	()
Operating conditions:	
Core voltage:	1.2V
Junction temperature range:	0-85 °C

*Répertoire du projet*

< Back    Next >    **Finish**    Cancel    Help

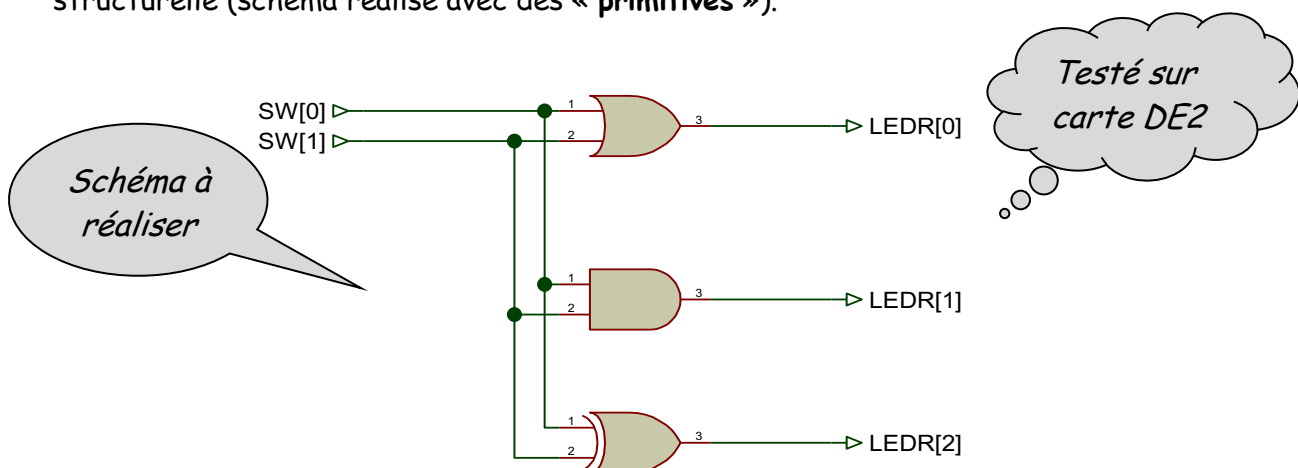
Après vérification **Cliquer** sur « **Finish** ». Sinon **corriger** !

Si tout est OK après avoir cliqué sur « Finish » la fenêtre du premier plan apparaît et est mise à jour :



### B) Description schématique d'un circuit (très) simple.

Avant de « coder » en VHDL un circuit nous allons commencer par le décrire sous forme structurée (schéma réalisé avec des « primitives »).



Répondre aux questions sur votre compte-rendu.

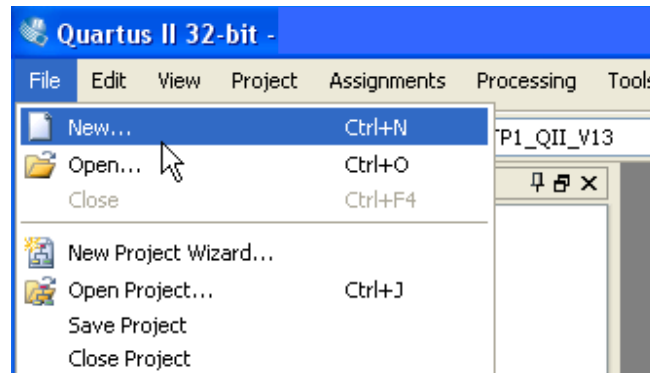
**Question1** : a) Quel est le nom de ces portes logiques ?

b) Donner leur table de vérité.

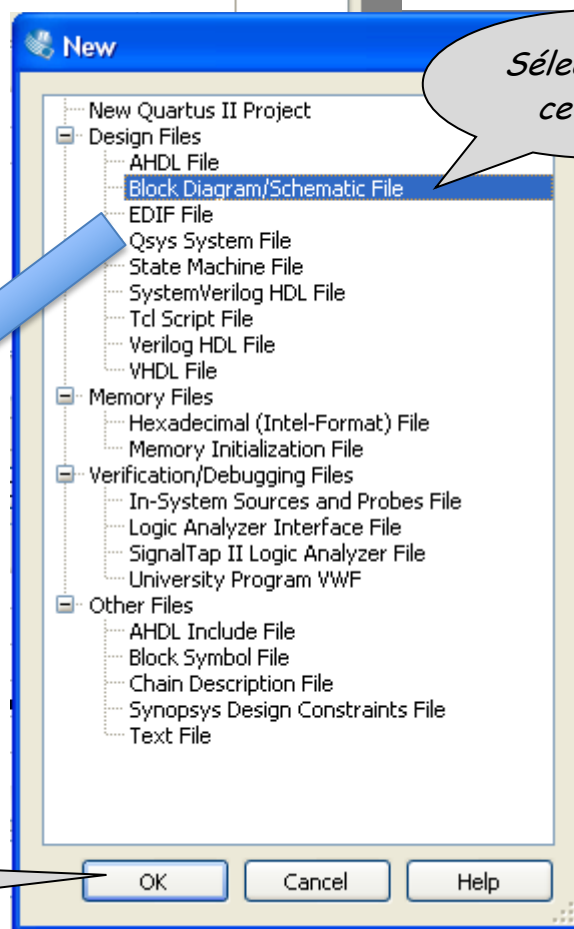
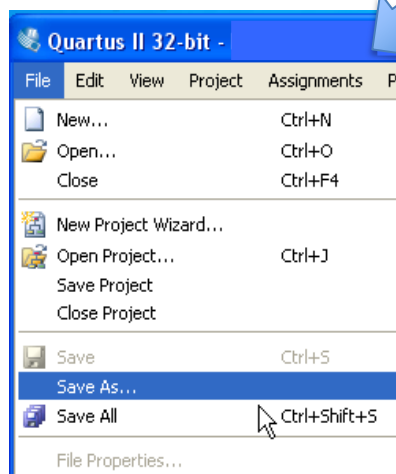
→ Faire valider par votre enseignant.

✚ **Étape N°1 : Créer un nouveau fichier le Nommer et le Sauver**

- Suivre pour cela les actions ci-dessous :



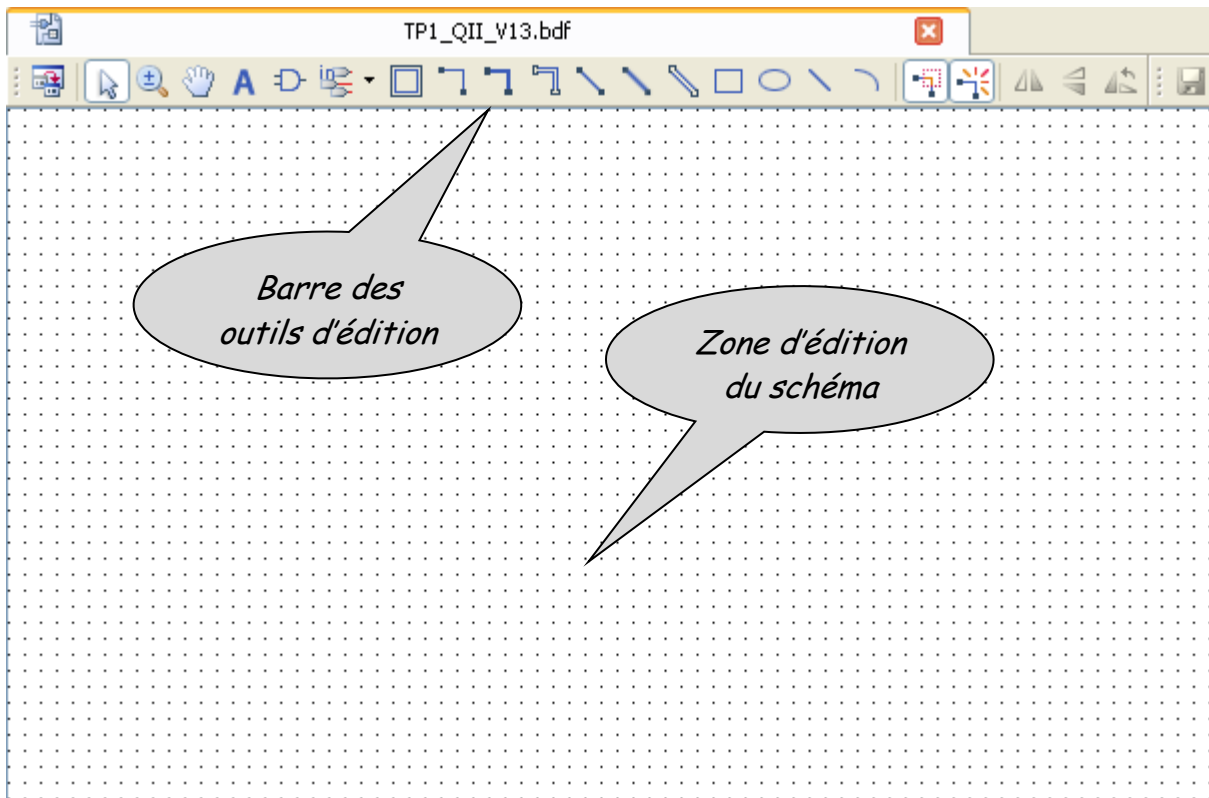
La fenêtre ci-contre s'ouvre:  
La fenêtre d'édition du schéma  
s'ouvre. Le nom par défaut est  
**block1** d'extension **bdf** pour  
block diagram file.



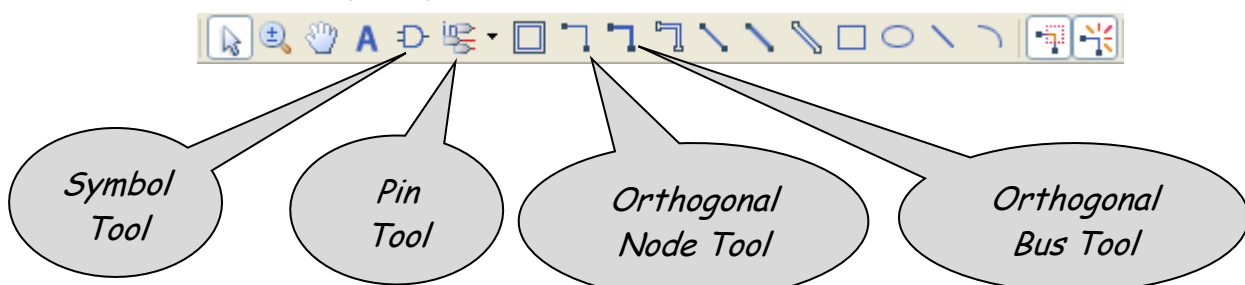
- Renommer ce fichier en **TP1\_QII\_V13**



La fenêtre d'édition de schémas est réactualisée !

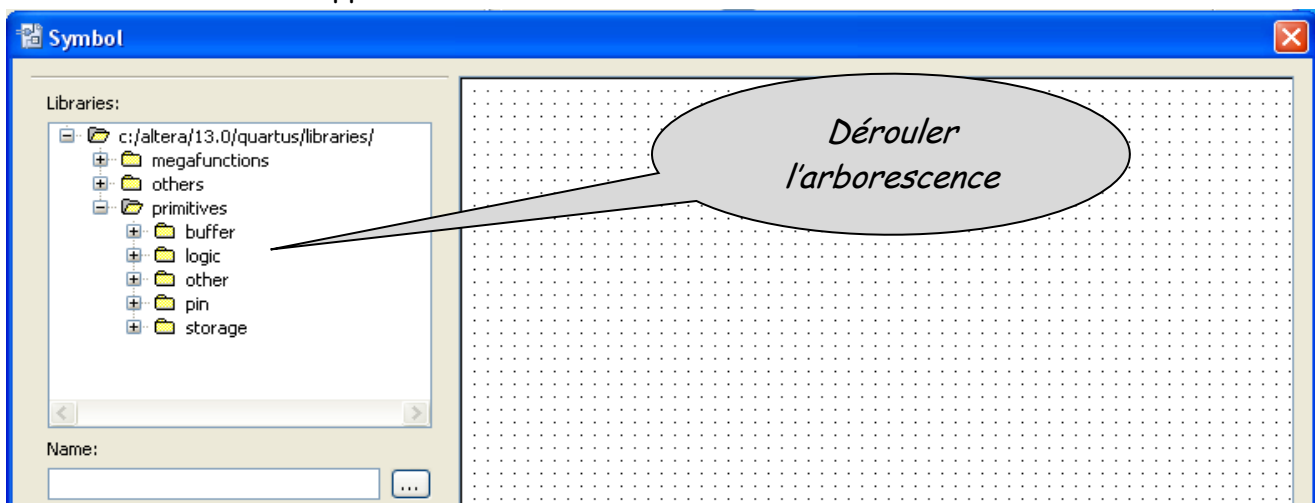


Noms des principaux boutons que nous allons utiliser :



➤ **Étape N°2 : Dessiner le schéma structurel.**

- Choisir le bouton **Symbol Tool** ou cliquer 2x dans la zone d'édition de schéma, vous verrez apparaître la fenêtre ci-dessous :





- Choisir *primitive*. Vous avez accès à tous les éléments logiques de base.

Entre autres :

- Le dossier *logic* contient toutes les portes logiques *and*, *or*, *nand*, etc,
- Le dossier *pin* contient le type d'E/S, *bidir*, *input*, *output*.

- Sélectionner une porte ET à 2 entrées (and2) et cliquez sur OK.

Vous verrez que l'objet sélectionné est attaché à votre souris.

- Cliquer pour placer l'objet à l'endroit où est situé le curseur.

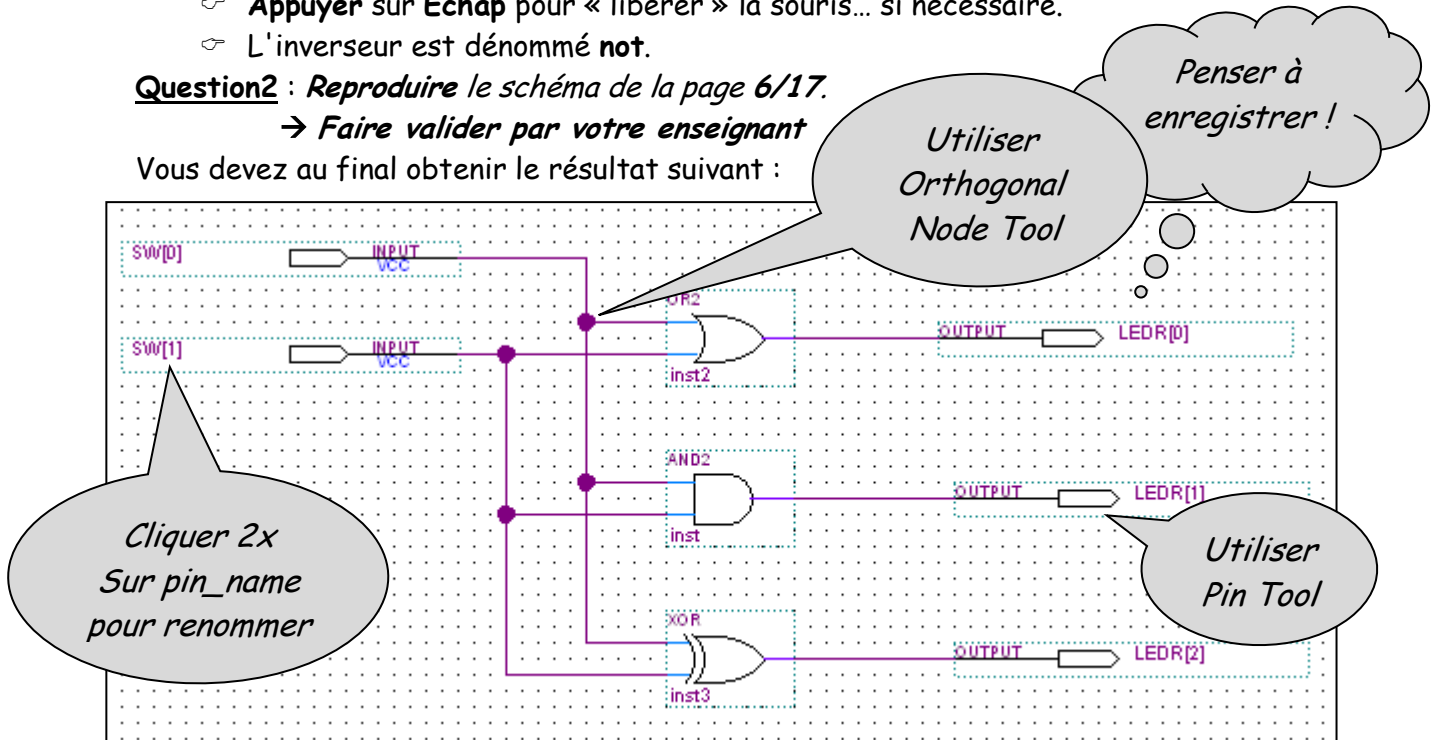
☞ Appuyer sur Echap pour « libérer » la souris... si nécessaire.

☞ L'inverseur est dénommé *not*.

**Question2 : Reproduire le schéma de la page 6/17.**

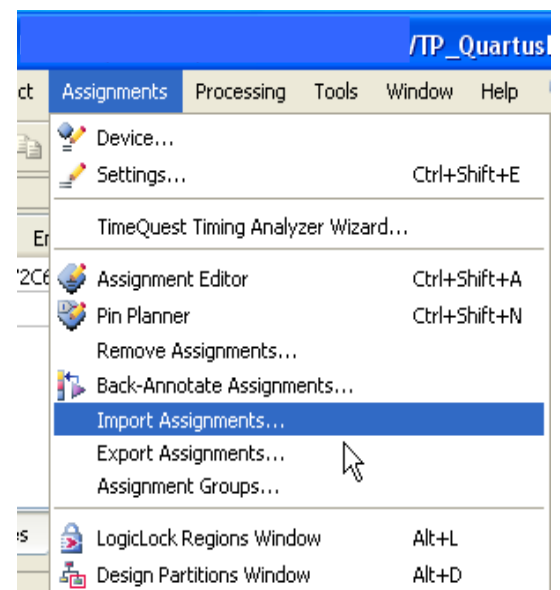
→ Faire valider par votre enseignant

Vous devez au final obtenir le résultat suivant :

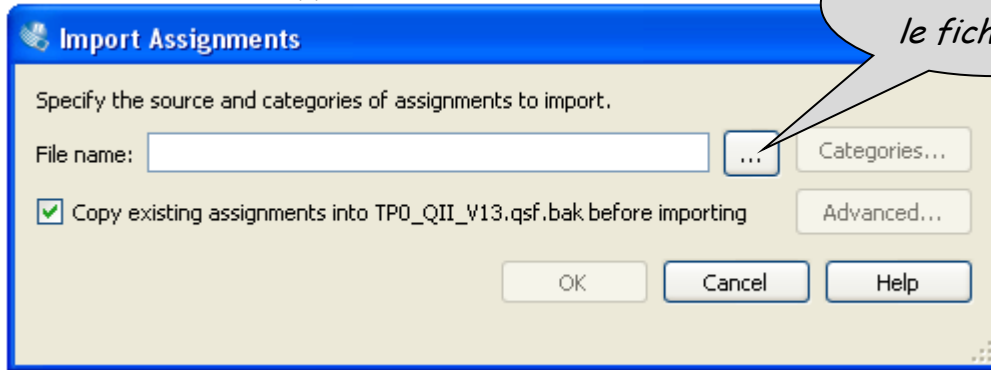


### Étape N°3 : Assignment des broches

Afin d'associer le **nom réel** des broches du FPGA (*PIN\_AE23* par exemple) à des noms plus évocateurs (comme *LEDR[0]*), pouvant être reconnus par le logiciel **Quartus II**, il faut importer le fichier *DE2\_pin\_assignments.csv*. Pour cela dérouler le menu **Assignments** et choisir l'option **Import Assignments** :



La fenêtre ci-dessous apparaît :

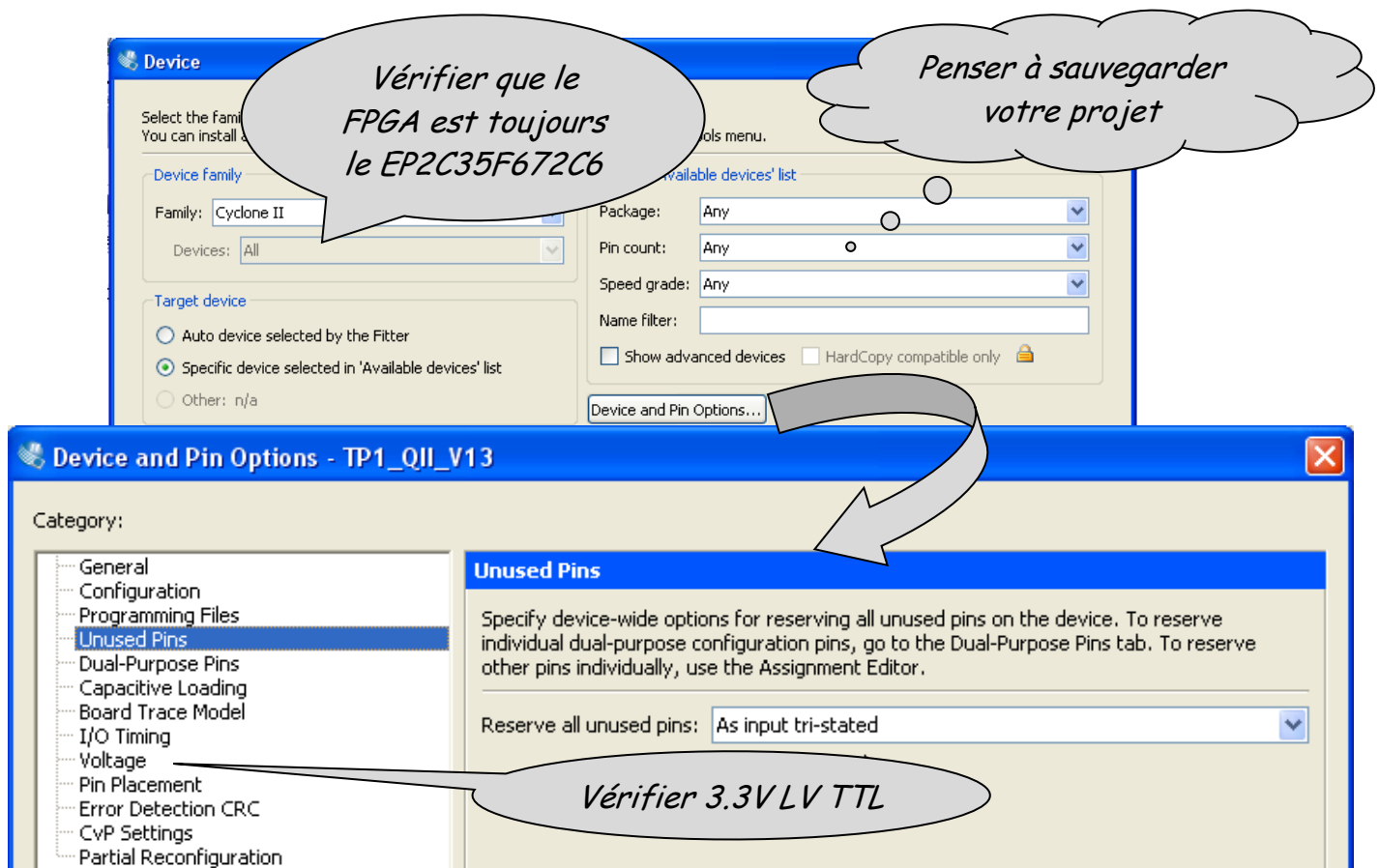


- **Spécifier** ensuite l'emplacement du fichier dans **File name**, (Voir Dossier Ressources\_DE2). La correspondance des PINs du FPGA doit apparaître immédiatement. Les broches (pins) qui ne seront pas utilisées dans votre montage devront être placées en mode **As input tri-stated**.

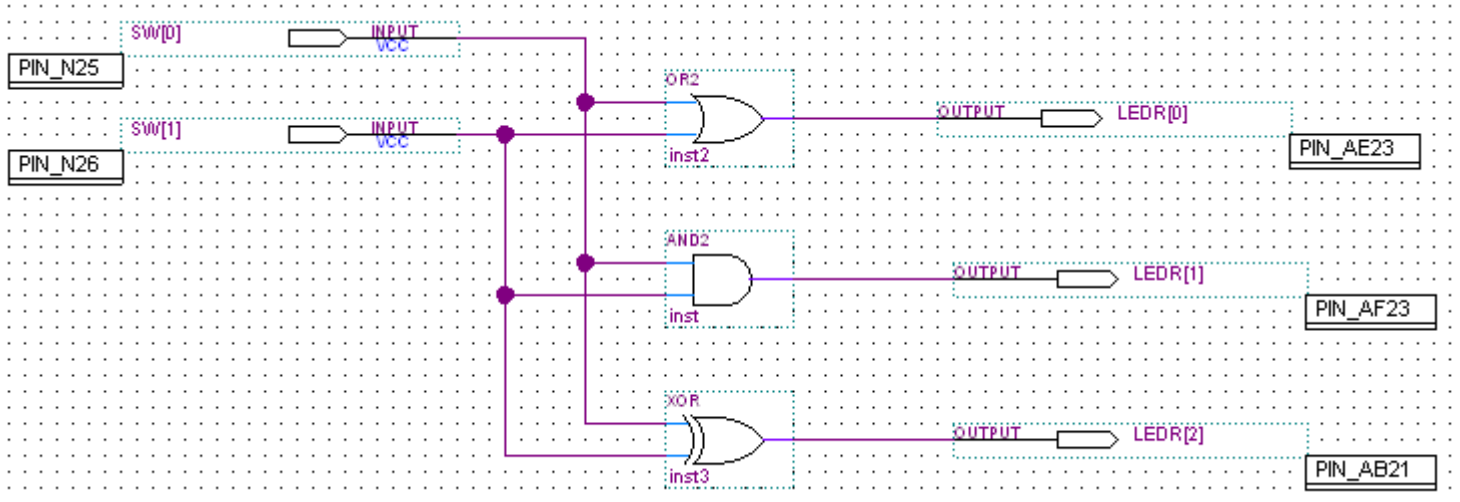
Pour cela aller dans le menu **Assignments Device** puis

- **Cliquer** sur le bouton **Device and Pin Options**,
- **Choisir** la catégorie **Unused Pins** et sélectionner **As input tri-stated** dans le même déroulant. Voir les copies d'écran des fenêtres page suivante.

**Le non-respect de cette étape se traduira  
par une exclusion du TP ainsi que par l'attribution  
d'une note de 0/20 car risque de destruction du FPGA !!**



En fin d'édition votre schéma doit ressembler à ceci :



→ Faire valider par votre enseignant.

**Question3** : Rédiger dans votre compte-rendu une description VHDL de type « flot de données » du circuit ainsi réalisé. Respecter les noms donnés aux signaux d'E/S.

Le nom de la boîte noire (entity) sera : **schema\_1\_TP1**

Le nom de la vue interne (architecture) sera : **Flot\_schema\_1\_TP1**

→ Reprendre le cours si nécessaire.

### C) Compilation du projet.

La compilation est un processus permettant de transformer la description d'un circuit (description schématique dans notre cas) en un fichier de "programmation" permettant de configurer (programmer) le FPGA pour implémenter matériellement le circuit décrit. Pour un petit projet comme celui-ci, le processus devrait prendre moins de 30 secondes, mais pour des circuits plus importants, le processus de compilation peut nécessiter plus de 30 minutes !

🔧 **Étape N°1** : Lancement du processus de compilation.

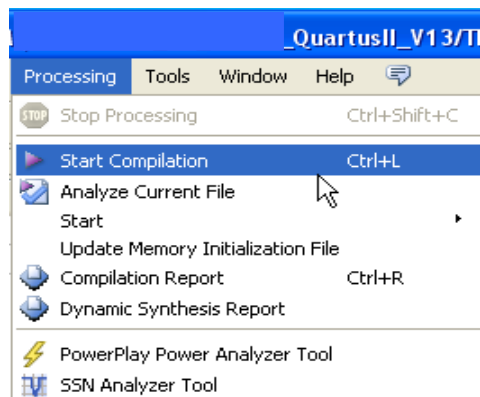
Choisir dans Quartus  
Flow : Compilation



Dans le menu **Processing** choisir **Start Compilation**. ...ou appuyez le bouton



de la  
barre d'outils.

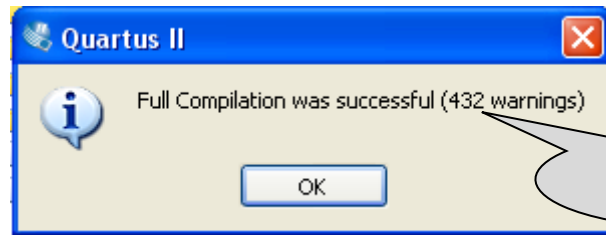


## Étape N°2 : Corriger les erreurs !

À la fin du processus de compilation, une fenêtre vous indiquera s'il y a des erreurs dans votre circuit. Si tel est le cas, vous pouvez vous référer aux alertes dans la fenêtre des messages en bas de l'écran.

Vous pouvez « **double-cliquer** » sur le premier message d'erreur (rouge) pour que l'outil vous indique la source du problème rencontré.

Une erreur fréquente consiste à court-circuiter deux sorties (sorties de portes logiques). Deux sorties sont court-circuitées si elles sont reliées par un même fil.

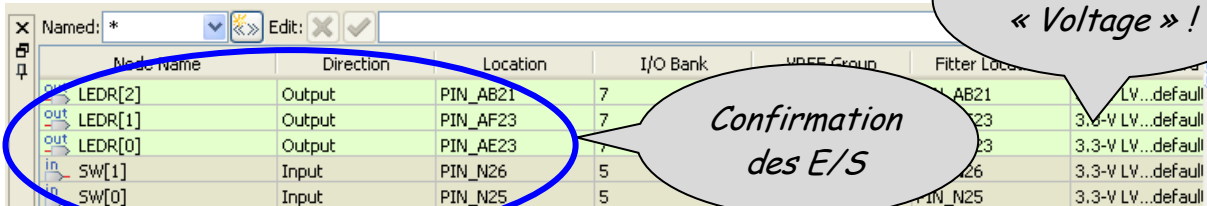


C'est normal !

Sinon, si vous avez seulement des avertissements (warnings), ne pas en tenir compte !

## Étape N°3 : Vérification de l'assignation.

Pour cela dans le menu **Assignments** choisir l'option **Pin Planner**.



Named	Mode Name	Direction	Location	I/O Bank	VPFF Group	Fitter Location	Filter Location
out	LEDR[2]	Output	PIN_AB21	7		AB21	LV...default
out	LEDR[1]	Output	PIN_AF23	7		F23	3.3-V LV...default
out	LEDR[0]	Output	PIN_AE23	7		E23	3.3-V LV...default
in	SW[1]	Input	PIN_N26	5		N26	3.3-V LV...default
in	SW[0]	Input	PIN_N25	5		N25	3.3-V LV...default

Puis **Vérifier** alors l'assignation (voir votre schéma) comme indiquée ci-dessus.

## D) Création automatique d'une description VHDL à partir d'un schéma.

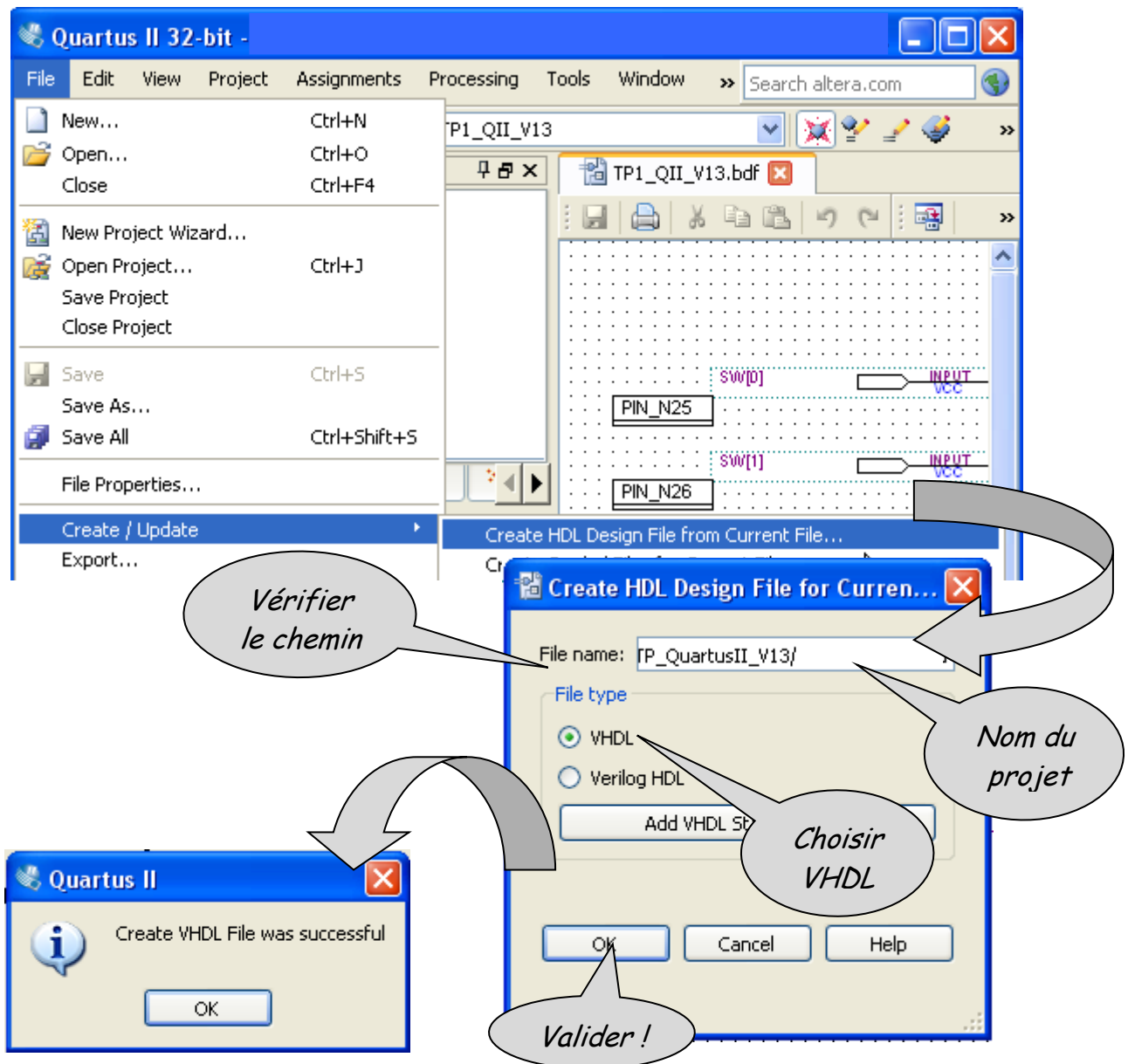
L'outil **Quartus II** permet la conversion automatique vers une description VHDL.

Vérifier que le « focus » dans le navigateur de projet est bien sur **TP1\_QII\_V13.bdf**

## Étape N°1 : Créer un fichier VHDL.

Dans le menu **File** choisir l'option **Create/Update**

- Sélectionner **Create HDL Design File from Current File...**



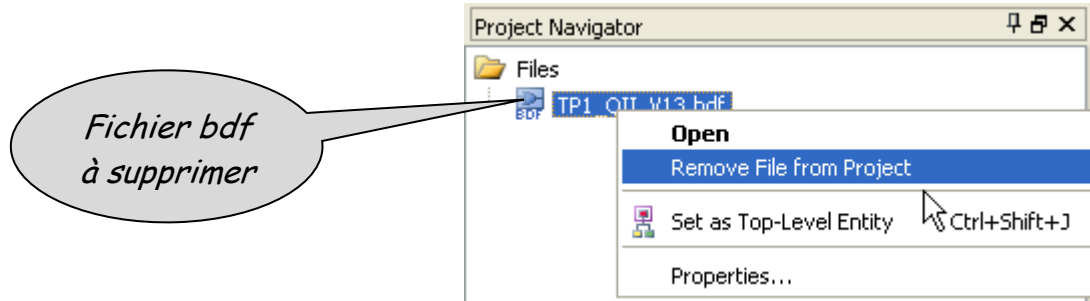
Nous avons maintenant une description VHDL TP1\_QII\_V13.vhd, ainsi qu'une description schématique TP1\_QII\_V13.bdf d'un même circuit dans le répertoire de projet.

**Attention**, un projet Quartus II ne peut contenir qu'une seule description à la fois d'un même circuit.

## Étape N°2: Supprimer/Ajouter un fichier du projet.

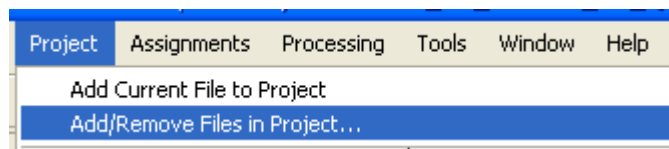
Pour supprimer le fichier **TP1\_QII\_V13.bdf** du projet,

- Aller dans l'onglet **File** du navigateur de projet,
- Faire un clic droit sur le fichier **TP1\_QII\_V13.bdf**
- Choisir l'option **Remove File from Project** du menu contextuel.

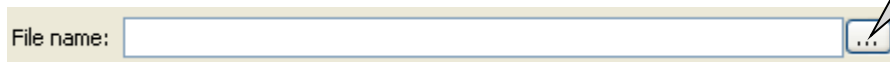


Pour **ajouter** le fichier **TP1\_QII\_V13.vhd** au projet, dans le menu :

- Aller dans le menu **Project**
- Choisir l'option **Add/Remove Files in Project...**



Dans le champ de saisie **File name**, de la fenêtre de Settings indiquer le répertoire dans lequel se trouve **TP1\_QII\_V13.vhd** ou rechercher le !



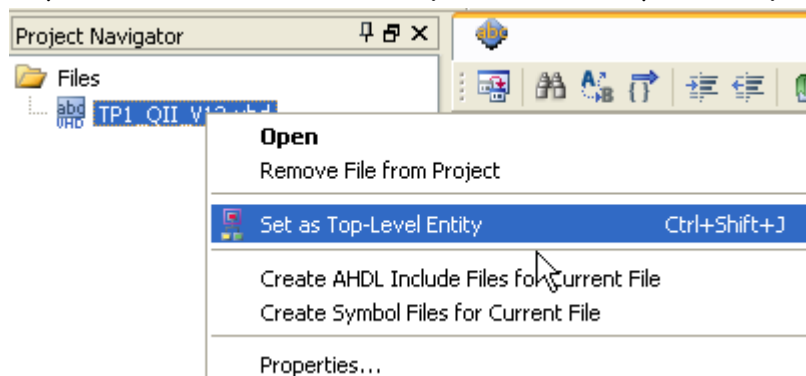
- Appuyer sur le bouton **Add** avant de valider par **OK** dans la fenêtre de Settings.

→ **TP1\_QII\_V13.vhd** doit être présent dans la liste de fichiers du projet du navigateur.

**Ce fichier doit correspondre à l'entité de haut niveau (Top Level Entity).**

Pour cela :

- Faire un clic droit sur le fichier **TP1\_QII\_V13.vhd**
- Choisir l'option **Set as Top-Level Entity**.
- Relancer la compilation dans Quartus II. Reprendre les étapes de la partie C si nécessaire.



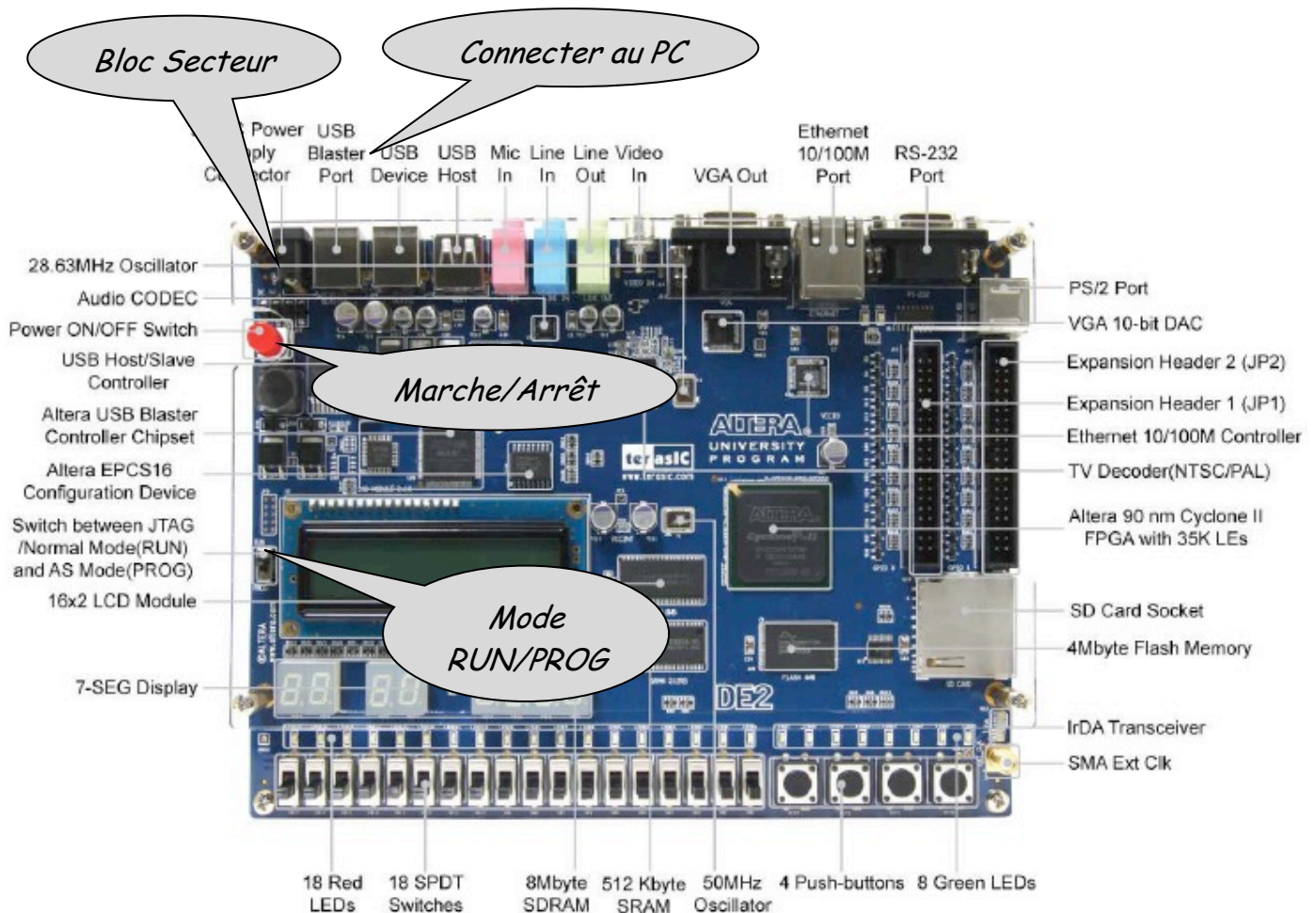


- **Question4** : *Cliquer 2 fois sur le fichier TP1\_QII\_V13.vhd. Analyser le fichier VHDL proposé par Quartus II. Comparer-le au votre édité à la Question3.*

### E) Programmation du FPGA de la carte DE2.

Attention la carte DE2 est FRAGILE et coûte chère !

#### 🚦 Étape N°1 : Connexion de la carte DE2



Lors de la mise en marche :

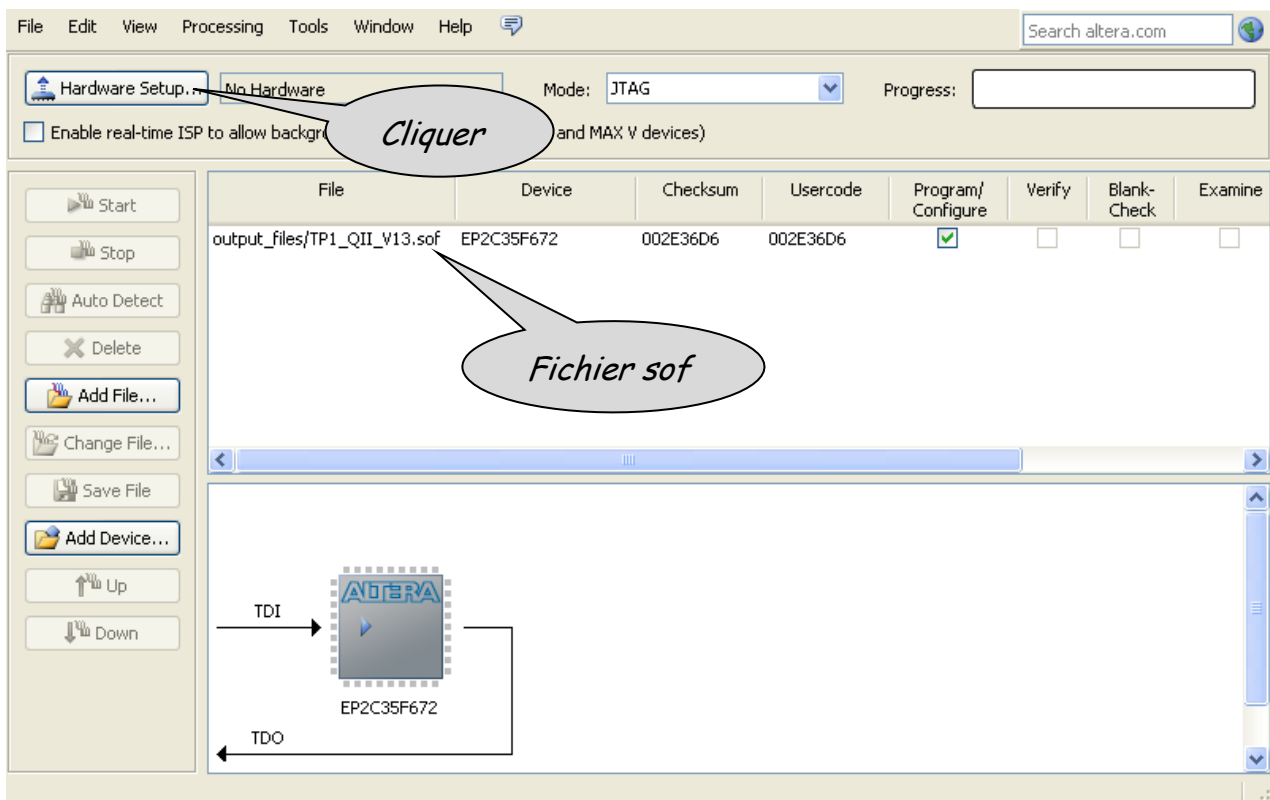
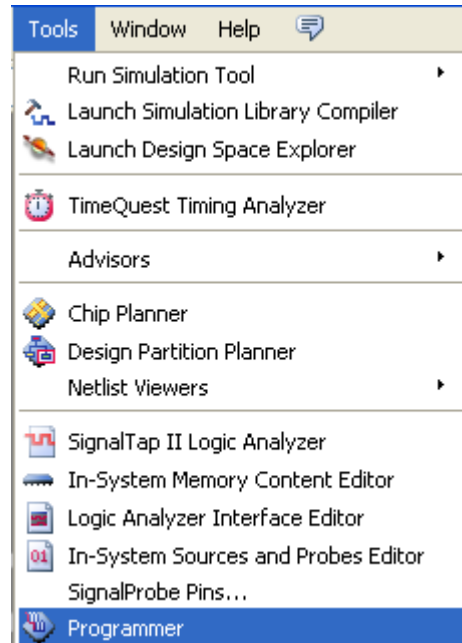
- La carte affiche un message de bienvenue sur l'afficheur LCD et un programme de démo tourne,
- Le pilote du port USB Blaster est normalement pris en charge par Windows 7\*,
- **Vérifier** que le switch Mode RUN/PROG est sur RUN,
- **Repérer les SW[0] et SW[1] ainsi que LEDR[0], LEDR[1], LEDR[2],**  
\*Demander à votre enseignant le cas échéant.



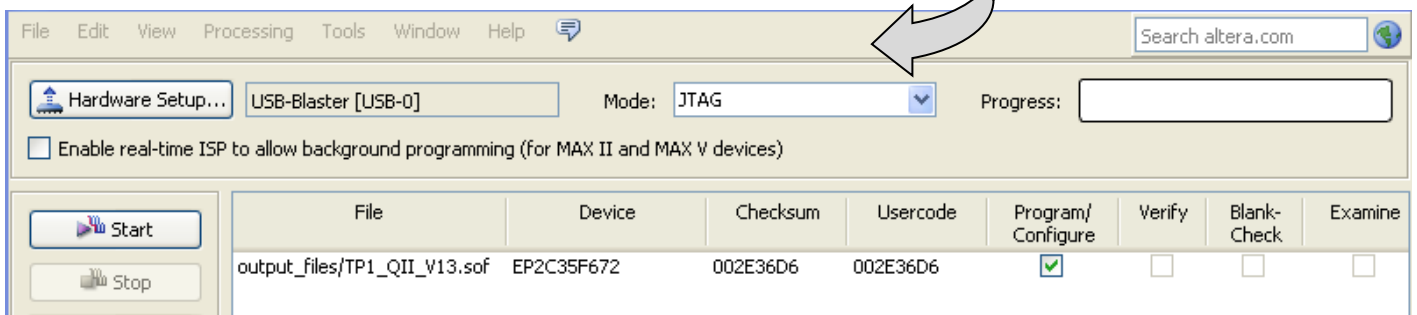
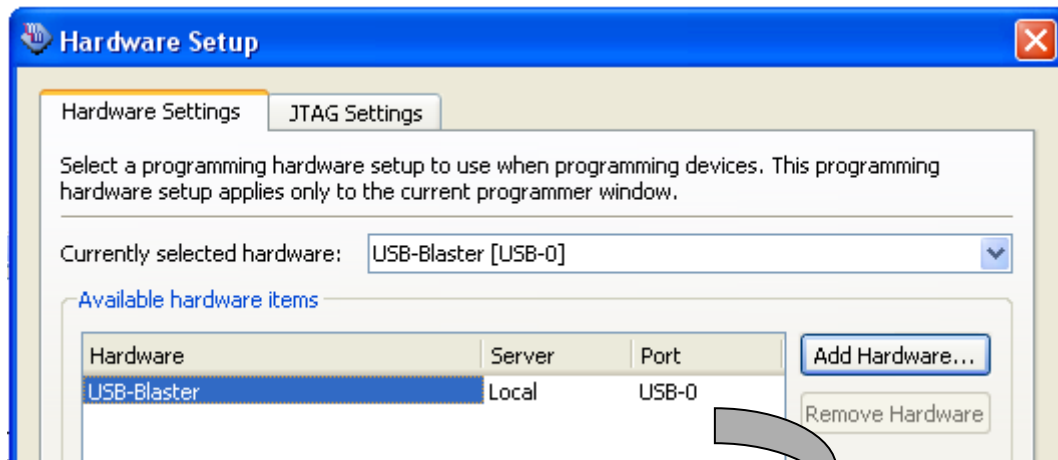
## Étape N°2 : Programmation du FPGA

Si toutes les étapes de compilation du projet se sont bien déroulées Quartus II a créé un fichier **TP1\_QII\_V13.sof** (sram object file). **Rechercher** ce fichier dans votre répertoire de projet.

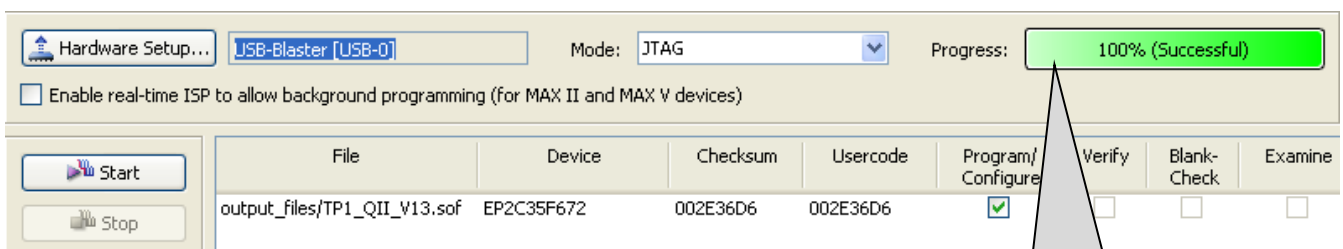
- Aller dans le menu **Tools** et choisir l'option **Programmer**,



- Rechercher le USB-Blaster [USB-0] dans la fenêtre *Currently selected hardware*
- Cliquer sur *Add Hardware*.



- Cliquer sur *Start*.



- Le programme « démo » a laissé sa place au notre.
- **Tester** le circuit étudié.
- **Faire valider par votre enseignant.**

Fin de la première partie  
du TP N°1.

N'oubliez pas d'envoyer  
votre CR