

IFT-1227 Demo 1

Philippe Gabriel

<2022-09-16 Fri>

Contents

1	Installation Quartus	1
1.1	Vieux Guide	1
1.1.1	Compte Intel	2
1.1.2	Fichiers à télécharger - Installation	2
1.1.3	Installation sur gnu/linux (facultative)	2
1.1.4	Configurations de variables	2
1.1.5	Créer un projet	4
1.1.6	Ajouter des fichiers au projet	4
1.1.7	Exemple avec Block Diagram	4
1.2	Nouveau Guide	5

1 Installation Quartus

Voir le nouveau guide pour les détails d'installation sur Windows

- Il est fortement recommandé d'installer Quartus sur une machine Windows
- Si vous voulez tenter de le télécharger sur gnu/linux, vous pouvez tenter de suivre les étapes dans le vieux guide mais avec le fichier d'installation de la nouvelle version de Quartus
- Si vous êtes sur MacOS, il vous faut une machine virtuelle (préférentiellement Windows) pour installer Quartus

1.1 Vieux Guide

Lien de téléchargement windows et gnu/linux. Pour Mac utiliser une machine virtuelle.

1.1.1 Compte Intel

Il faut se créer un compte standard avant de pouvoir installer

- Remplir les champs obligatoires
- Voir son courriel pour un email de vérification

1.1.2 Fichiers à télécharger - Installation

Télécharger ces fichiers dans le même **répertoire**

- Sélectionner Quartus II Software V13.0 sp1 sous *Web edition*
- Dans *Device Families*, cocher Cyclone II, Cyclone III, Cyclone IV et ModelSim-Altera STARTER edition (Free)

Run le installer (processus plus "facile" sur Windows)

1.1.3 Installation sur gnu/linux (facultative)

Pour gnu/linux, s'assurer de changer la permission avec cette command:
`chmod +x *.run` Important d'avoir installer cette librairie sur ubuntu si
votre système est 64-bit: `sudo apt install libc6-i386` Puis rouler la com-
mande à partir de l'emplacement des fichiers d'installations: `./QuartusSetupWeb-13.0.1.232.run`
Skip le manuel puis arriver à ces informations ici - S'assurer de bien répondre
aux détails d'installation:

Message	Répo
Quartus II Web Edition (Free) [Y/n]	y
Quartus II Web Edition (Free) - Quartus II Software (includes Nios II EDS) (4424MB) [Y/n]	y
Quartus II Web Edition (Free) - Quartus II Software 64-bit support (1090MB) [Y/n]	y
Quartus II Web Edition (Free) - Devices [Y/n]	y
Quartus II Web Edition (Free) - Devices - Cyclone II/III/IV (615.2MB) [Y/n]	y
ModelSim-Altera Starter Edition (Free) (3547.1MB) [Y/n]	y
ModelSim-Altera Edition (3547.1MB) [y/N]	n
Is the selection above correct? [Y/n]	y

1.1.4 Configurations de variables

Dans votre fichier `~/.profile`, ajouter ceci à la fin

```
export QUARTUS_ROOTDIR="/home/phil-gab99/Packages/Quartus/altera/13.0sp1/quartus"
PATH="$PATH:$QUARTUS_ROOTDIR/bin"
```

Dans votre fichier `~/.bash_aliases`, ajouter ceci à la fin (facultatif)

```
alias quartus='quartus --64bit'
```

Installer les paquets suivants supplémentaires pour quelques parties dans Quartus qui fonctionnent toujours en 32-bit

```
sudo dpkg --add-architecture i386
sudo apt update
sudo apt install libc6:i386 libncurses5:i386 libstdc++6:i386 libpng12-0:i386
libfreetype6:i386 libsm6:i386 libxrender1:i386 libfontconfig1:i386
libxext6:i386 libxtst6:i386 libxi6:i386 libgtk2.0-0:i386
```

(Note: il se peut que `libpng12-0:i386` ne soit pas disponible sur votre distro, utiliser plutôt ce lien et recherché ce nom spécifique: `libpng12-0_1.2.54-1ubuntu1.1+1~ppa0~eoan_amd64.deb` pour l'installer manuellement avec `sudo dpkg -i`)

Finalement, pour permettre à votre système de bien reconnaître la carte, voir les instructions à ce lien

Créer un fichier (une règle) à `etc/udev/rules.d/92-usbblaster.rules`
- Fonctionne pour la plupart des distros

```
# USB-Blaster
SUBSYSTEM=="usb", ATTRS{idVendor}=="09fb", ATTRS{idProduct}=="6001", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="09fb", ATTRS{idProduct}=="6002", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="09fb", ATTRS{idProduct}=="6003", MODE="0666"
```

```
# USB-Blaster II
SUBSYSTEM=="usb", ATTRS{idVendor}=="09fb", ATTRS{idProduct}=="6010", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="09fb", ATTRS{idProduct}=="6810", MODE="0666"
```

Pour Ubuntu 12.04/14.04/16.04 créer un fichier (une règle) à `etc/udev/rules.d/51-usbblaster.rules`

```
# USB Blaster
SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTR{idVendor}=="09fb", ATTR{idProduct}==

SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTR{idVendor}=="09fb", ATTR{idProduct}==

SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTR{idVendor}=="09fb", ATTR{idProduct}==

# USB Blaster II
SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTR{idVendor}=="09fb", ATTR{idProduct}==

SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", ATTR{idVendor}=="09fb", ATTR{idProduct}==
```

Vous aurez besoin des super user rights (**su**) pour ajouter une règle

1.1.5 Créer un projet

1. Create a New Project NEXT
2. Sélectionner l'emplacement d'installation et le nom du projet NEXT
3. Passer la prochaine étape NEXT
4. Choisir la carte approprié - Inscrit sur la carte, mais devrait normalement être **Cyclone II - EP2C35F672C6** NEXT
5. Passer la prochaine étape NEXT
6. FINISH

1.1.6 Ajouter des fichiers au projet

Vous pouvez maintenant créer des fichiers au projet (block diagram schéma, fichiers .vhd, etc...)

1.1.7 Exemple avec Block Diagram

Ajouter des **inputs** et **outputs**, leur donner des noms, **Symbol Tool** pour ajouter les différentes portes logiques

Enregistrer et compiler:

- Bouton **Save**

- Bouton **Start Compilation**

Quand tout fonctionne comme prévu, on veut maintenant entreprendre l'assignation avec la carte physique:

- Bouton **Assignment Editor**
- Associé les différentes composantes aux pins de la carte (spécifié dans les énoncés normalement) - Un checkmark indique que tout est bon pour cette ligne
 - Le **Assignment Name** prend la valeur de **Location**

Après avoir complété les assignations pour les **inputs** et **outputs**, enregistrer et recompiler pour s'assurer que tout est correct.

- Bouton **Programmer** sélectionner le fichier **.sof** (dans le répertoire **output_files** si n'est pas automatiquement sélectionné)
- Bouton **Hardware setup** pour vérifier qu'il reconnaît la carte (si elle est connectée à votre ordinateur) - Entrée **USB Blaster**
- Après avoir ajouté la carte, le bouton **Start** va permettre de charger le circuit sur la carte - Assurer vous que la carte est allumée
- Vous pouvez maintenant expérimenté en flipant les switchs appropriés

1.2 Nouveau Guide

Quartus Download

- Tab *Intel Max Series*
- Configuration *Intel Max Series 10 FPGA*
- Version Lite selon l'OS utilisé (préféablement Windows, utilisé une VM sinon)
- Download la version *21.1.0.842-windows.tar*

Ou accéder au lien directement <https://www.intel.com/content/www/us/en/software-kit/684216/intel-quartus-prime-lite-edition-design-software-version-21.html>

```
$ tar --version # S'assurer que l'outil tar existe
```

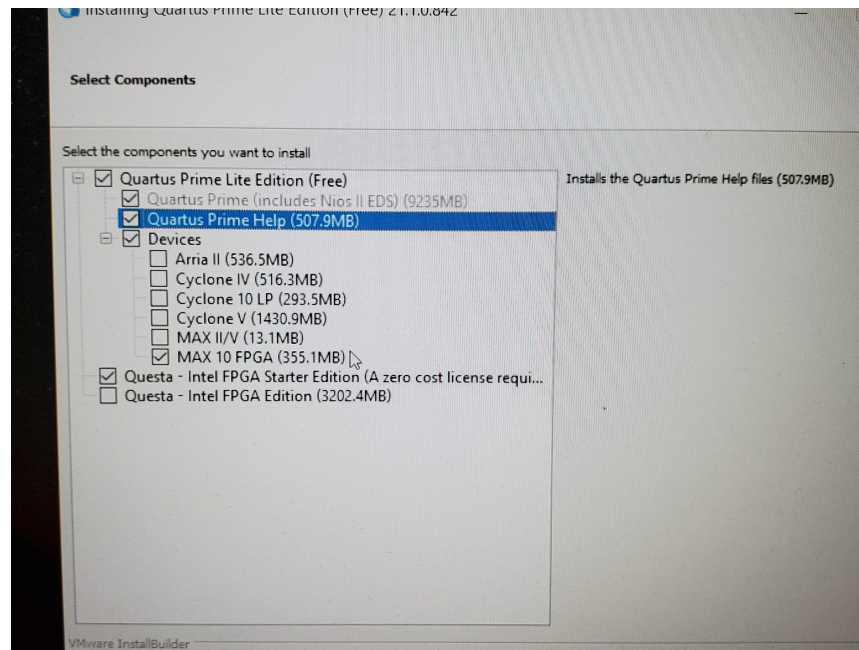
La commande ci-dessus devrait afficher un output de la version de tar utilisé. Si ça affiche une erreur, il faut installer l'outil **tar** en ligne (par exemple ici) Par la suite, naviguer au répertoire d'installation

```
$ tar -x Quartus-lite-21.1.0.842-windows.tar
```

Rouler le **setup.bat**

```
$ .\setup.bat
```

Sélectionnez les options suivantes



Après avoir compléter l'installation, vous pouvez tenter de créer un projet et d'expérimenter avec Quartus en suivant des étapes similaires à partir de la section du vieux guide *Créer un Projet*

Plus de détails à la prochaine démo sur l'utilisation de Quartus